NON-LINEAR MODEL PREDICTIVE ENERGY MANAGEMENT STRATEGIES FOR STAND-ALONE DC MICROGRIDS

ARASH MORADINEGADE DIZQAH

PhD

2014

NON-LINEAR MODEL PREDICTIVE ENERGY MANAGEMENT STRATEGIES FOR STAND-ALONE DC MICROGRIDS

ARASH MORADINEGADE DIZQAH

A thesis submitted in partial fulfilment
of the requirements of the
University of Northumbria at Newcastle
for the degree of
Doctor of Philosophy

Research undertaken in the
Faculty of Engineering and Environment

February 2014

Abstract

Due to substantial generation and demand fluctuations in stand-alone green microgrids, energy management strategies (EMSs) are becoming essential for the power sharing purpose and regulating the microgrids voltage. The classical EMSs track the maximum power points (MPPs) of wind and PV branches independently and rely on batteries, as slack terminals, to absorb any possible excess energy. However, in order to protect batteries from being overcharged by realizing the constant current-constant voltage (IU) charging regime as well as to consider the wind turbine operational constraints, more flexible multivariable and non-linear strategies, equipped with a power curtailment feature, are necessary to control microgrids.

This dissertation work comprises developing an EMS that dynamically optimises the operation of stand-alone dc microgrids, consisting of wind, photovoltaic (PV), and battery branches, and coordinately manage all energy flows in order to achieve four control objectives: i) regulating dc bus voltage level of microgrids; ii) proportional power sharing between generators as a local droop control realization; iii) charging batteries as close to IU regime as possible; and iv) tracking MPPs of wind and PV branches during their normal operations.

Non-linear model predictive control (NMPC) strategies are inherently multivariable and handle constraints and delays. In this thesis, the above mentioned EMS is developed as a NMPC strategy to extract the optimal control signals, which are duty cycles of three DC-DC converters and pitch angle of a wind turbine.

Due to bimodal operation and discontinuous differential states of batteries, microgrids belong to the class of hybrid dynamical systems of non-Filippov type. This dissertation work involves a mathematical approximation of stand-alone dc microgrids as complementarity systems (CSs) of Filippov type. The proposed model is used to develop NMPC strategies and to simulate microgrids using Modelica.

As part of the modelling efforts, this dissertation work also proposes a novel algorithm to identify an accurate equivalent electrical circuit of PV modules using both standard test condition (STC) and nominal operating cell temperature (NOCT) information provided by manufacturers. Moreover, two separate stochastic models are presented for hourly wind speed and solar irradiance levels.

To my Family

”It always seems impossible until it is done.”

Nelson Mandela

Contents

List of Tables

List of Figures

Nomenclature

ϵ	Relaxation parameter of the regularized equivalent to a MPCC problem
$\Gamma(.)$	Vector of the condition functionals of a mixed complementarity problem
$\gamma(.)$	Vector of the slack variables of a mixed complementarity problem
λ_{opt}	The nominal tip speed ratio of a wind turbine that generates the maximum power coefficient
$\mathcal{C}$	Convex cone
$\mathcal{C}^*$	The polar cone of a convex cone $\mathcal{C}$
$\mathcal{F}$	A vector of implicit DAE functionals
$\mathcal{G}$	Vector of inequality functionals in an OCP
$\mathcal{H}$	Hamiltonian function
$\mathcal{L}$	Lagrangian function
$\mathcal{L}$	Stage cost of an OCP
$\mathcal{MCP}$	Mixed complementarity problem
$\mathcal{M}(.)$	Stabilizing terminal penalty term of an OCP_N
$\mathcal{P}$	Equality constraint of an OCP
$\mathcal{R}$	Vector of boundary conditions in an OCP
$\mathcal{U}$	Permissible range of control variables
$\mathcal{X}$	Permissible range of differential states
$\mathcal{Z}$	Permissible range of algebraic states

∇_z Gradient matrix of a functional on variable z

∇_{zz} Hessian matrix of a functional on variable z

Ω Stabilizing terminal region constraint of an OCP_N

$\perp$ Complementarity operator

ρ Penalty parameter of the penalized equivalent to a MPCC problem

C_p Power coefficient of a wind turbine

C_p The maximum power coefficient of a wind turbine

D_s Duty cycle of the connected boost converter to the solar branch

I_0 Reverse saturation current of a diode

I_{batdc} The output dc current of the battery branch of a dc microgrid (A)

I_{load} Load demanded dc current of a dc microgrid (A)

I_{ph} Photocurrent of a PV module (A)

I_{pvdc} The output dc current of the solar branch of a dc microgrid (A)

I_{pv} The PV array current (A)

$I_{sc,stc}$ Short-circuit current at STC (A)

I_{wtdc} The output dc current of the wind branch of a dc microgrid (A)

$J_{infty}(.)$ Cost function of an OCP over an infinite horizon

$J_N(.)$ Cost function of an OCP over an finite horizon N

K Boltzman constant (1.38066×10^{-23})

k_I The short-circuit current temperature coefficient ($A/^\circ C$)

k_V Open-circuit voltage temperature coefficient ($V/^\circ C$)

N_s The number of PV cells in PV module (-)

OCP_{infty} An OCP with infinite horizon

OCP_N An OCP with finite horizon of N

q Electron charge (1.60218×10^{-19})

R_L The connected linear load to a dc microgrid (Ω)

S The current solar irradiance (W/m^2)

S_{stc} Insolation level (W/m^2)

$T_{c,stc}$ STC cell temperature (K)

T_c The current PV cell temperature (K)

$u^*(.)$ Optimal solution of an OCP

V_{dc} DC bus voltage of a dc microgrid (V)

$V_{infty}(.)$ Value function of an OCP over an infinite horizon

$V_N(.)$ Value function of an OCP over an finite horizon N

$V_{oc,stc}$ Open-circuit voltage at STC (V)

V_{pv} The PV array voltage (V)

Acknowledgements

Nobody conducts his/her research in an isolated environment. I benefited from a number of remarkable advisors, colleagues, and friends during the past three years of my life and research at Northumbria University.

Foremost, I would like to express my deep gratitude to my principal supervisor, Dr. Alireza Maheri, for his all encouragements and constructive advices. He has been a patient supporter throughout the research and I admire him for his trust.

I also deeply thank Prof. Krishna Busawon, as my second supervisor. Prof. Busawon was always quick to grasp what I had done and was very supportive throughout the research. He is an example of passion and excellence in research.

Furthermore, I am particularly thank Prof. Peter Fritzson at Linköping University, Sweden for his helpful comments and assistance on using OpenModelica throughout this research. The application of Modelica modelling language in long-term simulation and optimal controllers grew directly out of the researches being done by his research group at Linköping University.

For this work, I was supported by Synchron Technology Ltd., which is greatly appreciated. Foremost, Mr. Taj Nezami from Synchron Technology Ltd. who has long been there to listen and to offer sound advices. This work is also supported by Northumbria University which is gratefully acknowledged.

Arash Moradinegade Dizqah
Newcastle upon Tyne, UK
June 12, 2014

Declaration

I declare that the work contained in this thesis has not been submitted for any other award and that it is all my own work. I also confirm that this work fully acknowledges opinions, ideas and contributions from the work of others. The work has done in collaboration with Synchron Technology Ltd, UK.

Any ethical clearance for the research presented in this thesis has been approved. Approval has been sought and granted by the University Ethics Committee on 20 February 2014.

Name:

Arash Moradinegade Dizqah

Signature:

Date:

June 12, 2014

Chapter 1

Introduction

This thesis deals with developing energy management strategies (EMSs) that dynamically optimise the operation of stand-alone dc microgrids, consisting of wind, photovoltaic (PV), and battery branches, to coordinately manage energy flows and regulate dc bus voltage level of microgrids. The proposed strategies are novel constrained and multivariable maximum power point trackers (MPPTs) that employ renewable energy systems as flexible generators which means that their generated powers are optimally curtailed, if required, in proportion to their ratings.

The presented EMSs are developed using non-linear model predictive control (NMPC) technique and dynamically extract the optimal control signals with respect to the battery bank and wind turbine operations. In order to develop these NMPC-based strategies, this dissertation work also involves the mathematical modelling of stand-alone dc microgrids as a complementarity system (CS) of Filippov type which is applicable to both the NMPC strategies, as well as for the long-term simulation purpose.

As parts of the modelling efforts, a stochastic model is also presented to simulate hourly wind speed and solar irradiance for locations across the UK. Furthermore, a novel algorithm is proposed to identify an accurate equivalent electrical circuit for PV modules.

The remainder of this chapter reviews the notion of sustainable microgrids and briefly describes three wind, solar, and battery branches. A discussion on energy management strategies is then provided which is followed by an overview of the

NMPC technique to implement multivariable and non-linear optimal controllers for the constrained problems. The chapter is closed with a review on research contributions.

Chapter 2 provides mathematical modelling of stand-alone dc microgrids which is shown to be of the class of hybrid dynamical systems due to connected battery bank. The discontinuous differential states and bimodal operations are modeled as separate complementarity constraints in order to approximate stand-alone dc microgrids as systems of Filippov type.

Chapter 3 investigates stochastic models of renewable energy resources. In order to design a reliable site and a robust controller for stand-alone sustainable microgrids, uncertainties in renewable energy resources need to be modeled stochastically. The goodness-of-fit of the proposed model has been evaluated using Kolmogorov-Smirnov (K-S) test.

Chapter 4 addresses the problem of accurate model identification of the available physical components across sustainable microgirds and in particular photovoltaic (PV) modules. A novel PV model identification method is presented which is fast and accurate regardless the underlying PV technologies.

Chapter 5 presents the developed SIMULINK and Modelica models to simulate operation of dc microgrids. SIMULINK/SimPowerSystems is used to model and simulate the fast and very fast dynamics of dc microgrids. Modelica model, on the other hand, covers slow dynamics of the system for NMPC applications and long-term simulation purpose.

Chapter 6 proposes novel energy management strategies to address three control issues of dc microgrids; i.e. voltage regulation, power sharing, and battery management. The EMSs proposed in this work simultaneously control four variables: i) wind turbine pitch angle; ii) angular velocity of wind generator; iii) operating voltage of PV array; and iv) charging current of battery bank.

Several concluding remarks and future research directions are presented in Chapter 7.

1.1 Stand-alone dc microgrids

Microgrids are new key elements of modern power grids that improve the grids capability of hosting renewable energy and distributed storage systems [1]. In fact, in the near future, distribution networks will consist of several interconnected microgrids that will locally generate, consume, and even store energy [1]. Microgrids may operate as an extension of the main grid, i.e. the grid-connected mode, or as a stand-alone network with no connection to the grid. Stand-alone sustainable microgrids have some distinct applications in avionic, automotive, or marine industries, as well as in remote rural areas. In such stand-alone microgrids, intermittent solar and wind energies coupling with battery storages contributes realistic sources to supply variable load demands [2]. However, comparing to the grid-connected microgrids, three well-known issues regarding voltage regulation, power sharing, and battery management, are more severe in stand-alone microgrids leading to the necessity of more sophisticated control strategies.

Although traditionally the utility grids have always been of ac type, there are several reasons for employing dc microgrids for small-scale rural areas or commercial facilities [1, 3–7]. While ac systems benefit from the ease of transmission, transformation, distribution, and protection [6], they suffer from the need of synchronization of several generators [3]. On the other hand, dc microgrids are more efficient due to the facts that dc generators and storages have taken higher portion of modern grids capacity [1] and there is no need of ac-dc converters to connect them to dc microgrids [7].

Figure 1.1 illustrates the topology of a stand-alone dc microgrid for small-scale applications. It consists of wind, solar, and battery branches which are connected to a dc bus through dc-coupled structures, i.e. via dc-dc converters. The microgrid supplies a variable linear dc load which is connected directly to the grid bus.

From Figure 1.1, it can be seen that the presented dc microgrid is controlled by four manipulated variables, i.e. the wind turbine pitch angle and switching duty cycles of three different dc-dc converters. While increasing the wind turbine pitch angle promotes pitching to feather, the operating points of PMSG, PV, and battery bank can be changed by varying dc-dc converters duty cycles.

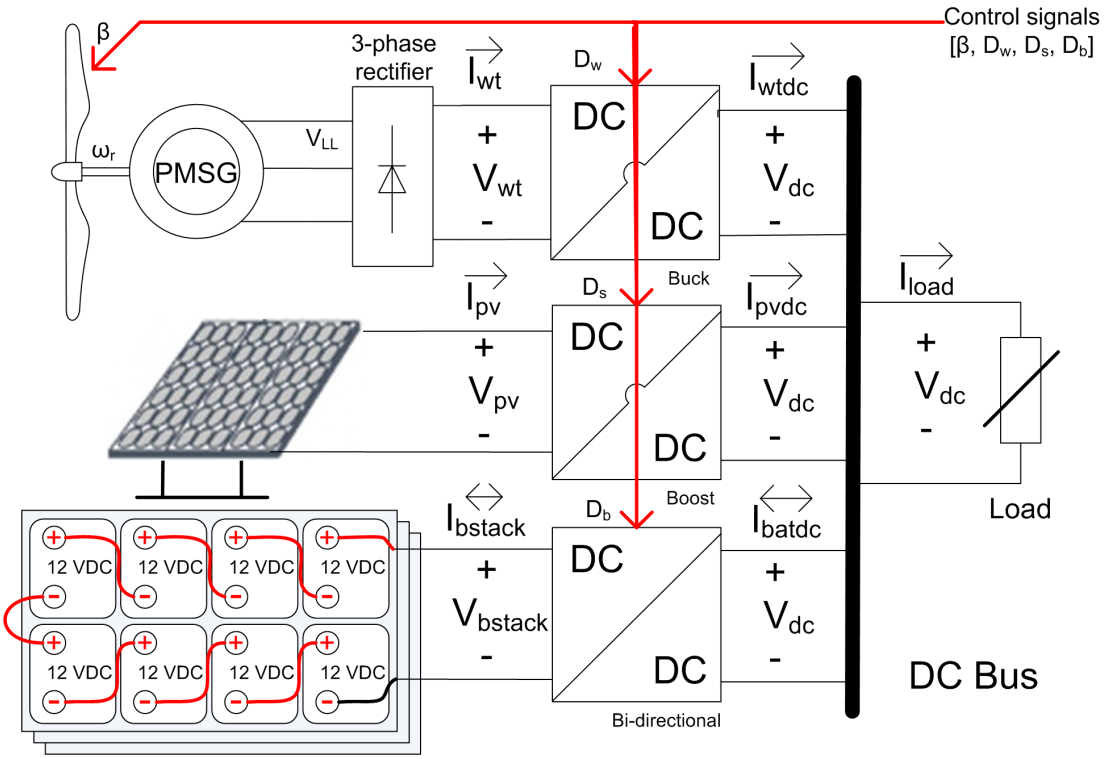


Figure 1.1: Topology of stand-alone green microgrids in this study

1.1.1 Wind turbines

Wind turbines (WTs) convert the kinetic energy of wind to mechanical power [8]. The performance of a WT can be characterized with three different curves, namely, power, torque, and thrust coefficient curves [9]. These curves are normally plotted in terms of tip speed ratio, which is defined as a weighted ratio of the rotational speed to wind speed, for different values of pitch angle. Figure 1.2 illustrates the optimum value of tip speed ratio at which the power coefficient, C_p, is maximized. According to the Lanchaster-Betz theory, the upper bound of power coefficient, i.e. $C_{p,max}$, is 0.593 [9]. Modern wind turbines provide the maximum power coefficient of around 0.48 [9].

Regarding the rotational speed, a WT operation can be classified into either constant speed or variable speed. The variable speed WT operation, which requires employing power converters, is more attractive, principally due to its ability to harvest the maximum power at variable wind speeds [10]. In order to generate the maximum power by a WT at variable wind speed, it is necessary to employ a maximum power point tracking (MPPT) control strategy [11].

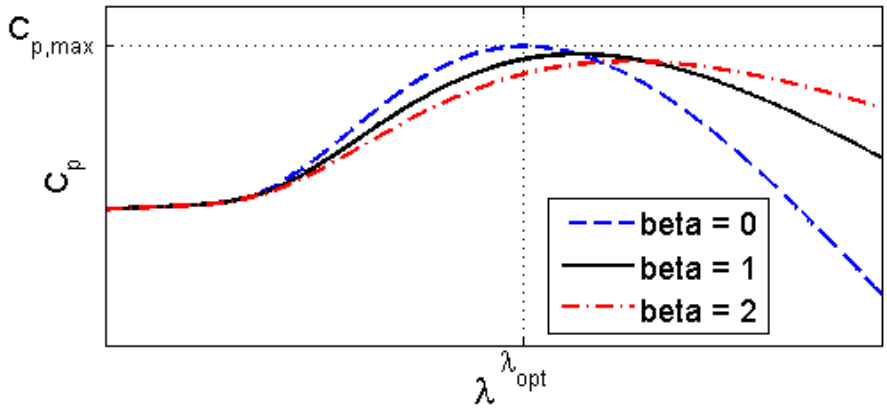


Figure 1.2: Power coefficient curves in terms of tip speed ratio for three different values of pitch angle.

1.1.2 Permanent magnet synchronous generators (PMSGs) and three-phase rectifiers

A wind turbine can be connected to an electrical generator directly or through a gear-box. The former case, which is called direct-drive, provides some advantages in terms of high reliability and is more popular for small-scale wind turbines [12]. To deliver power in direct-drive topology, which operates at low rotational speed, it requires employing multi-pole generators. In spite of high cost, PMSGs are the most dominant type of direct-drive generators in the market [12], due to several advantages such as higher efficiency.

In order to convert the three-phase output of a PMSG to dc voltage, it is essential to deploy a three-phase rectifier. A general structure, which consists of a full-bridge diode rectifier connected in series to a dc-dc converter, is common, due to lower cost [11]. Figure 1.3 illustrates the electrical circuit of a full-bridge three-phase rectifier where I_{wt} is the equivalent dc current of dc side. Due to the inductance L_s available on ac side, there is non-instantaneous current commutation [13].

1.1.3 Boost, buck, and bi-directional dc-dc converters

dc-dc converters are normally implemented based on the switching-mode circuit technology containing at least one energy storage and a transistor-based power-pole [13]. However, in ideal cases, a single-pole double-throw switch can also be used for simulation purpose [14]. Figure 1.4 illustrates the electrical circuit of boost, buck, and bi-directional types of dc-dc converters. While a boost-type converter, as given in Figure 1.4a, scales up its input voltage, a buck-type converter, as given in

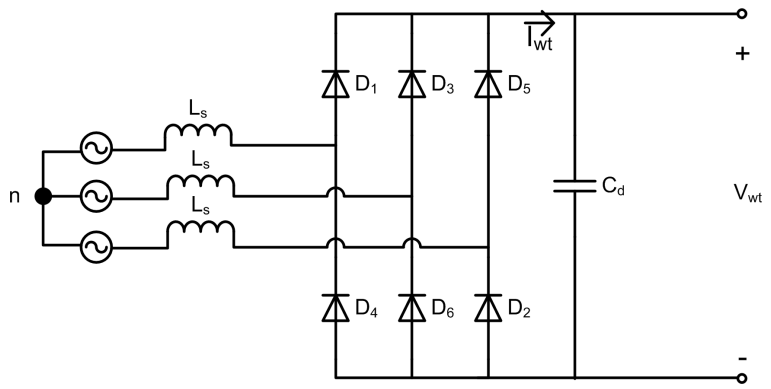


Figure 1.3: Electrical circuit of a full-bridge three-phase rectifier.

Figure 1.4b, provides lower voltage than the input voltage. Unlike the boost and buck converters which dictate the instantaneous current flow to be unidirectional, Figure 1.4c shows a bi-directional converter. In such a converter a complementary control signal allows the current to flow in either direction.

A state-space averaging approach to model a dc-dc converter is proposed in [15]. It suggests two states, I_l and V_c for the continuous conduction mode (CCM) in which the instantaneous inductor current, I_l, is always greater than zero. According to the proposed approach, there is a set of two distinct state-space systems to model two states of switch operation and the overall state-space model of the converter is a weighted average of these two models. The weighting factor is the duration of time that converter remains in each state.

While state-space averaging approach is simple to analyse and implement, it does not model the hybrid nature of converters. One-level dc-dc converters given in Figure 1.4 work in two different modes of operation with respect to the value of discrete state S_d [16]. Defining the same state vector as above, i.e. $X^T = [I_l \quad V_c]^T$, dc-dc converters can be modelled as hybrid systems. It presents an affine state space model coupled with a linear output equation for each modes of operation [16].

1.1.4 Photovoltaic (PV) arrays

PVs are among the popular renewable energy components to harvest solar energy. A PV cell, as the fundamental PV element, is a P-N junction that converts solar irradiance to the electrical energy. Normally, manufacturers provide PV modules, also known as PV panels, which consist of several PV cells connected together in series. In order to construct a PV farm, at least one PV array, which is a combination

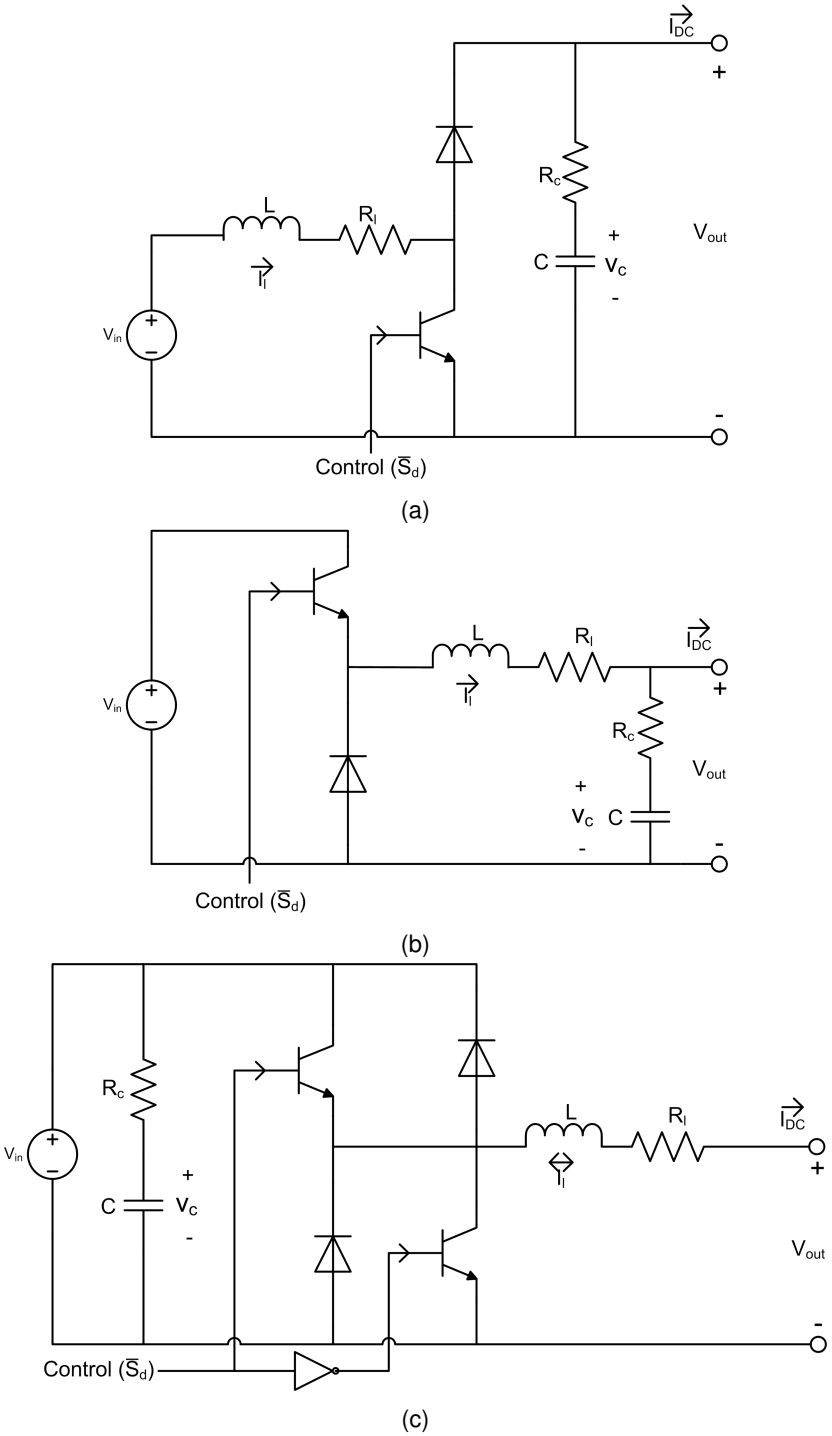


Figure 1.4: Ideal electrical circuit of (a) boost-type; (b) buck-type; and (c) bi-directional dc-dc converters.

of several PV modules in series and parallel arrangement, is used to provide the required power at a specific voltage level.

A PV cell is a non-linear component that its operation is characterised by a set of current-voltage curves at different insolation levels and junction temperatures. Figure 1.5 shows a sample of such current-voltage $(I - V)$ curves indicating the maximum power point (MPP). MPP varies due to changes in operating conditions, i.e. insolation level and junction temperature, and requires being followed by a MPPT algorithm.

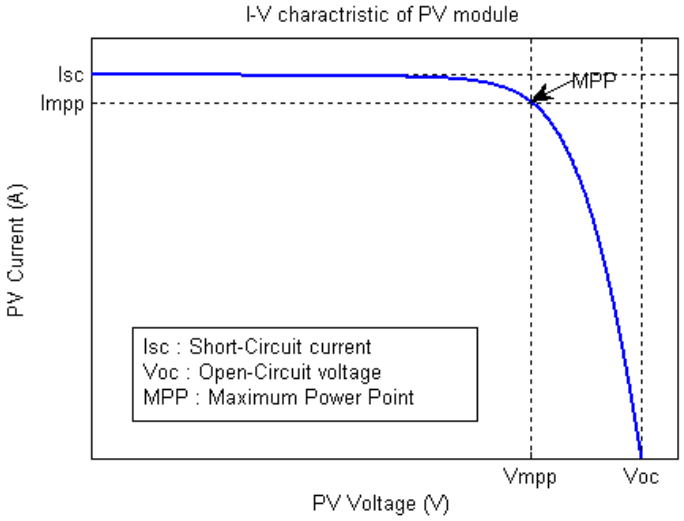


Figure 1.5: A sample $I - V$ curve of PV modules.

For further study on the PV technology, the reader is referred to Chapter 4.

1.1.5 Lead-acid batteries

There are different types of batteries applicable to the backup/storage purposes across microgrids. Different battery technologies are comprehensively reviewed and compared in [17, 18] for the renewable energy applications. They suggest that lead-acid batteries have some advantages for hybrid renewable energy system (HRES) applications. Lead-acid batteries are widely available in many sizes and are appropriate for small to large applications. Furthermore, the normalized cost of this type of batteries is reasonable and it is mature in concepts, mathematical model and technology. In fact, the performance characteristics of lead-acid batteries are well understood and modelled.

Batteries operate in different modes. Figure 1.6a illustrates a complete operating

cycle of a battery from charging to the exhaustion zone in which battery is completely discharged [19]. In Figure 1.6a, it can be seen that the battery is a multi-mode component introducing discontinuity to the system. An equivalent electrical circuit, as Figure 1.6b, and a hybrid mathematical formulation of lead-acid batteries for two charging and discharging modes of operation are proposed in [20].

Despite the power losses, it is argued in [21] that employing a dc-coupled structure to connect a battery bank to the dc bus is more flexible in terms of implementing different charging and discharging regimes.

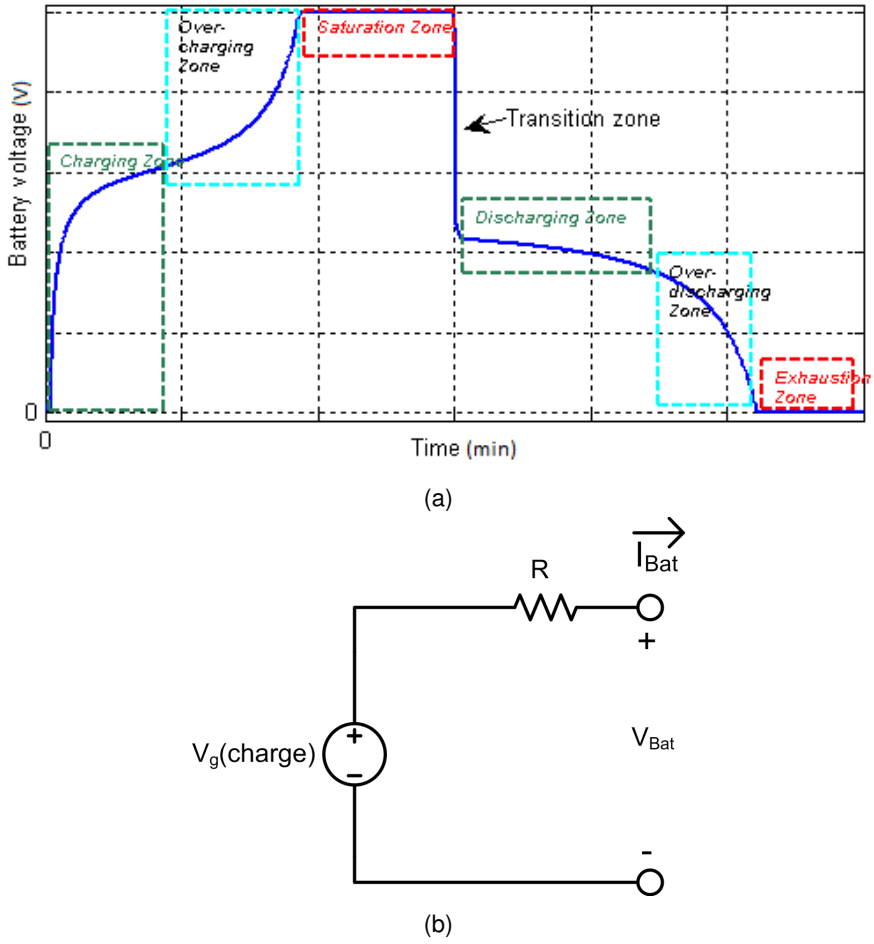


Figure 1.6: (a) Different working zones and (b) equivalent electrical circuit of lead-acid batteries.

1.2 Acausal versus causal modelling and simulation

The complex systems, like stand-alone dc microgrids described in Chapter 2, normally includes different operating modes which are also limited with algebraic constraints. As a result, such systems can be modelled, according to the acausal modelling approach, as a set of hybrid differential-algebraic equations (hybrid DAEs)

[22]. Unlike the causal approach, which needs the system to be decomposed into a chain of causal interacting blocks consisting of only ordinary differential equations (ODEs), acausal modeling is a declarative approach where the system model is described as hybrid DAEs [22]. Acausal modeling, as a mathematical representation of the system, is an effective way to model and simulate complex systems and such a model is also applicable to develop NMPC strategies. Modelica [23], as an object-oriented and equation-based language to describe complex systems, provides the capability to acausally model the class of hybrid dynamical systems [22].

1.2.1 Differential-algebraic equations (DAEs)

There are several engineering problems in which some algebraic states need to be considered in order to model a system. These algebraic states are often added by conservation laws, design constraints, or constitutive equations. For instance, though the movement of a dangling pendulum of length L can be modelled in Cartesian system with a set of ODEs, there is still an algebraic constraints by the length of pendulum which is $x^2 + y^2 = L^2$. Therefore, a dangling pendulum is modelled with a set of differential and algebraic equations (DAEs).

Differential-algebraic equations (DAEs) are a vector of functionals f in terms of both differential and algebraic states, denoted respectively by vectors x and z. Such a DAE system can be specified in the following semi-explicit non-linear form:

$$0 = f(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}; t), \quad \mathsf{x}(t_0) = \mathsf{x}_0, \tag{1.1a}$$

$$0 = p(\mathsf{x}, \mathsf{z}, \mathsf{u}; t). \tag{1.1b}$$

where u is a vector of the system inputs. It is important to note that the number of equations must equal the sum of the differential and algebraic variables, i.e. $dim(f) = dim(\mathsf{x}) + dim(\mathsf{z})$.

The structural properties of a system of DAEs can be described by a structure incidence matrix with rows of equations and columns of unknowns in any orders [24]. The S_{ij} element is set to be 1 if the ith equation contains the jth variable, else it is set to zero. Analysis of the structure incidence matrix leads to calculating a measure of

complexity for the system of DAEs, called as *index*. Generally speaking, the higher value the DAE index is, the more complicated the numerical solution routine.

In order to analyse the structure incidence matrix S, it is multiplied by a left and right permutation matrices P and Q, respectively:

$$\hat{S} = P.S.Q. \tag{1.2}$$

The resulting matrix $\hat{S}$, which has the same properties as the original matrix S, can be analysed as follows:

- If the resulting matrix $\hat{S}$ is a lower triangular matrix, the DAE index is zero which means that the DAE system can be converted into ODE form and there are neither algebraic loops nor structural singularities. In other words, there are explicit equations to calculate available algebraic states in terms of available differential states.

- If the resulting matrix $\hat{S}$ is a block lower triangular (BLT), the DAE index is one and it means that the system contains algebraic loops, but no structural singularities. A square matrix is BLT matrix if it can be divided into a number of building square matrices in which all blocks located upper the block diagonal are zero. The number and size of these algebraic loops are also defined by more deep analysis of the resulting BLT. In this case, algebraic loops must be solved separately by Newton-based solvers.

- If the resulting matrix $\hat{S}$ is singular, the DAE index is 2 or higher and the DAE system is called a higher index problem.

In order to solve DAE systems of higher index, one can reduce the index of the system by differentiating constraints. As a matter of fact, the index of a DAE system equals to the number of differentiation that lead to a full ODE system in which the algebraic constraints can also be replaced by equivalent ODE constraints. For instance, a dangling pendulum with length L can be modelled as the following Euler-Lagrange equations in the Cartesian coordinates:

$$\dot{x} = u, \tag{1.3a}$$

$$\dot{y} = v, \tag{1.3b}$$

$$\dot{u} = \lambda x, \tag{1.3c}$$

$$\dot{v} = \lambda y - g, \tag{1.3d}$$

$$x^2 + y^2 = L^2. \tag{1.3e}$$

where u and v are the momentum variables and λ, as a Lagrange multiplier, is the algebraic state. Twice differentiating the algebraic constraint in (1.3e) and replacing derivatives of the differential states and momentums with their equivalents, one obtains the following equation:

$$L^2\lambda - gy + u^2 + v^2 = 0. \tag{1.4}$$

The derivative of this equation leads to the following ODE constraint:

$$L^2\dot{\lambda} - 3gv = 0. \tag{1.5}$$

Therefore, Euler-Lagrange equations of a dangling pendulum in Cartesian coordinates are a DAE of index-3. However, in order to solve the higher index problems, the DAE system is differentiated index -1 times to be converted to a DAE of index-1. There are effective algorithms to solve the resulting DAE system of index-1 with algebraic loops [25].

1.2.2 Modelica modelling language

Although modelling and simulation of complex multidisciplinary systems is traditionally very important in all engineering disciplines, the area is expanded to develop more flexible models which are also applicable to model-based design methods or model-based control strategies. Modelica is an object-oriented declarative modelling language to model complex physical systems for different applications. The

language is continuously developed since 1997 and the last version of the specification, Modelica 3.3, was published in May 2012.

Modelica allows to declare the mathematical models of multidisciplinary complex systems by a unified modelling language to be used for simulation purposes or optimal control applications [26]. Modelica provides a way to model different systems by a set of mathematical equations which allow acausal modelling of multidisciplinary problems in the form of differential and algebraic equations, in a natural text-book style. The *connector*, defined by Modelica to model acausal physical interfaces, models a physical quantity by dedicated variables. The value and direction of these variables are given by the topology of the component network being fundamentally different from the causal models like Simulink ones [27].

Modelica models are not necessarily suitable for numerical algorithms or to be integrated. In order to use a developed Modelica model for simulation or optimisation purposes, it should be compiled to a form appropriate for numerical algorithms. Modelica platforms, such as OpenModelica [28] or Jmodelica [27], translate any developed model into a flat non-linear hybrid DAEs of index-1 representation consisting of variables and equations. Then, efficient programming codes, typically in C, are generated which are suitable for numerical integration [27]. These platforms support built-in general purpose index reduction algorithms, consistent initiator, general purpose DAE integrators, and probably a general purpose NLP. Therefore, Modelica models can be developed, compiled, and used for simulation or optimisation by employing Modelica platforms like OpenModelica or JModelica.

Tables 1.1 and 1.2 illustrate two simplified examples of Modelica codes. Table 1.1 shows a Modelica class to model lead-acid batteries. The bimodal operations of a lead-acid battery is modelled by a boolean variable, i.e. *chargeState*, which is *true* whenever the battery is in charging mode and *false* otherwise. Moreover, the differential state V_{exp} is initialized, using the *reinit* function provided by Modelica, when battery switches from discharging to charging mode of operation. The developed Modelica classes for lead-acid battery and other components can be employed to simulate a solar system consisting of a photovoltaic (PV) array connected to a load through a boost converter, as well as two batteries connected directly to the load to compensate the generation fluctuations. Table 1.2 summarizes a developed Model-

Table 1.1: An example of the lead-acid battery Modelica class.

class LeadAcidBattery

...

equation

$chargeState =$ **if** $noEvent(ibat \leq 0)$ **then** $true$ **else** $false$;

$der(charge) = 1/3600 * ibat$;

$der(Vexp) =$ **if** $chargeState$ **then** $P2/3600 * abs(i) * (P3 - Vexp)$ **else** $-(P2 * abs(i))/3600 * Vexp$;

when $change(chargeState)$ and $pre(chargeState)$ **then**

$tmp =$ **if not** $chargeState$ **then**

$pre(vbat) - V0 - R * pre(ibat) - (P6 * Cmax)/(Cmax - pre(charge)) * pre(charge) -$

$(P6 * Cmax)/(pre(charge) + 0.1 * Cmax) * pre(ibat)$

else 0;

$reinit(Vexp, tmp)$;

end when;

$soc = 1 - charge/Cmax$;

$vbat =$ **if** $chargeState$ **then**

$V0 - R * ibat - (P1 * Cmax)/(Cmax - charge) * charge$ - $(P1 * Cmax)/(charge + 0.1 * Cmax) * ibat + Vexp$

else $V0 - R * ibat - (P1 * Cmax)/(Cmax - charge) * charge - (P1 * Cmax)/(Cmax - charge) * ibat + Vexp$;

end LeadAcidBattery;

ica model for such a solar system. Since Modelica is an object oriented modelling language, it is possible to create multiple instances from a developed class. In Table 1.2, it can be seen that there are two instances of the lead-acid battery Modelica class that construct a battery bank by being connected together in series.

For further details on Modelica, the reader is referred to [26].

1.3 Hybrid dynamical systems

There are many dynamical systems that combine both continuous- and discrete-time behaviours. Such systems fit into the class of hybrid dynamical systems or simply hybrid systems. Generally speaking, hybrid systems are multi-modal systems that switch between modes when a particular event occurs [29]. These can be either state, time, or input originated events which are respectively generated by threshold crossing of a certain differential or algebraic state of the system, by the elapse of certain time period, or they may be generated externally. In each mode of operation, continuous states of the system are described by a set of DAEs which is changed by mode transitions. The mode transitions may also reset these states to

Table 1.2: An example Modelica model using the developed lead-acid battery Modelica class.

```
model SolarSystem
  Modelica.Blocks.Sources.Constant Sx(k = 300.0);
  Modelica.Blocks.Sources.Constant Tx(k = 298.15);
  Modelica.Electrical.Analog.Basic.Ground ground;
  PVArray pvarray(Npvp = 10);
  Modelica.Blocks.Sources.TimeTable DutyCycle(table = [...]);
  Modelica.Blocks.Sources.Step ramp(startTime = 5, offset = 10, height = -6);
  Modelica.Electrical.Analog.Basic.VariableResistor load;
  BoostConverter converter(Rl = 0.001, Rc = 0.3, L = 0.0005, C = 0.005);
  LeadAcidBattery battery1;
  LeadAcidBattery battery2;
equation
  connect(Tx.y,pvarray.Tx);
  connect(Sx.y,pvarray.Sx);
  connect(pvarray.n,ground.p);
  connect(pvarray.p,converter.p1);
  connect(converter.n1,ground.p);
  connect(converter.n2,ground.p);
  connect(DutyCycle.y,converter.D);
  connect(converter.p2,load.p);
  connect(ramp.y,load.R);
  connect(load.n,ground.p);
  connect(converter.p2,battery1.p);
  connect(battery1.n,battery2.p);
  connect(battery2.n,ground.p);
end SolarSystem;
```

new values. Such hybrid nature is naturally present in a variety of systems and is not necessarily caused by human intervention. For instance, the voltages and currents of an electrical circuit consisting of a switching device are changed continuously during each state of the switch; however, they are also varied discontinuously due to the switch operation.

A bouncing ball is another example of hybrid dynamical systems. Figure 1.7 shows a ball threw from height of $h_0 = 1m$ with the initial velocity of $V_0 = 0\ m/s$. From Figure 1.7, it can be seen that the motion direction and velocity are inverted at impact moment. In order to model such a hybrid system, such an impact is modelled with a discrete-time state; however, height and velocity states are continuous between each two successive impacts.

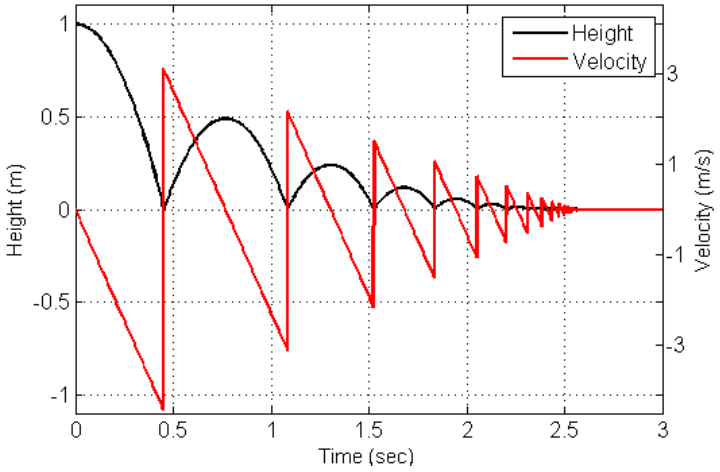


Figure 1.7: A bouncing ball as a hybrid dynamical system.

A model of hybrid dynamical systems requires describing both continuous- and discrete-time dynamics as well as regions in which these dynamics are applied to [30]. It should also take into account the interconnection between the continuous and discrete dynamics and mode transitions criteria. There are a number of approaches to model hybrid systems, including hybrid automata, differential inclusion, mixed logical dynamical systems, complementarity systems, and piecewise affine systems, to name a few. However, these different types of models can be, directly or under some conditions, transformed into each other [31]. All required conditions to establish such a transformation are related to well-posedness, i.e. the existence and uniqueness of solution trajectories given an initial condition, and boundedness of certain variables.

The hybrid automata approach can be used to model a large classes of hybrid dynamical systems. It explicitly partition states of a system into a continuous state ζ, including differential and algebraic states of the system, and a discrete state q, which describes different modes of operation. The discrete state q can only change through a jump. For instance, in order to model a bouncing ball given in Figure 1.7, a discrete state q must be defined that jumps from *normal* mode to *impact* mode of operation during the period of impact time. Figure 1.8 represents an example of a hybrid automaton with two modes of operation, q and q'. The evolved trajectories of the state ζ during modes q and q', are respectively defined by $\dot{\zeta} = f_q(q, \zeta)$, $\zeta \in Domain(q)$ and $\dot{\zeta} = f_{q'}(q', \zeta)$, $\zeta \in Domain(q')$. If the value of the state ζ is member of set $Guard(q, q')$, the system switches to mode q' and meanwhile ζ changes to

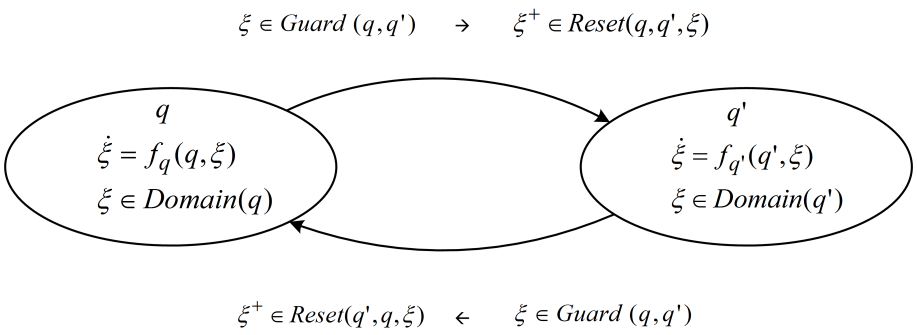


Figure 1.8: A simple example of an automaton.

a new value $\zeta^+ \in Reset(q, q', \zeta)$. For further details on the hybrid automata approach, the reader is referred to [30].

Although the hybrid autamata approach is very effective to model large classes of hybrid dynamical systems, it usually generates a complicated model in terms of analysis and controller design [29]. Indeed, for each of the above mentioned modelling approaches there is always a trade-off between generality of the model and complexity of the analysis. Among these approaches, differential inclusions, and particularly of Filippov type, is a straight-forward and natural approach for developing models to be used for computer aided simulation. Complementarity systems, on the other hand, is a preferable option to analyze and design controllers because it covers vast applications while it is agile enough in terms of theoretical investigations [31]. Moreover, there are prior studies and developed tools available for this approach. Due to these advantages, differential inclusions and complementarity systems are used in this study to model standalone dc microgrids.

For further study on hybrid dynamical systems analysis and modelling, the reader is referred to [29–31].

1.3.1 Complementarity systems (CSs)

CSs, as specific types of the class of hybrid dynamical systems, arise in several engineering applications dealing with multi-modal systems. A CS is formulated as the following general form which consists of orthogonal inequality (in (1.6c)) also known as complementarity problem [31]:

$$0 = \mathcal{F}(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{u}, \gamma; t), \tag{1.6a}$$

$$0 = \mathcal{G}(\mathsf{x}, \mathsf{u}, \gamma, \mathsf{w}; t), \tag{1.6b}$$

$$\mathcal{C}^* \ni \omega \perp \Gamma(\gamma) \in \mathcal{C}, \tag{1.6c}$$

$$x(t_0) = x_0. \tag{1.6d}$$

where $\perp$ is the complementarity operator indicating that at least one of the bounds is active. Let $\mathcal{C}$ be a convex cone, then $\mathcal{C}^*$ is the polar cone of $\mathcal{C}$ defined as $\mathcal{C}^* = \{z | z^T \nu \leq 0, \forall \nu \in \mathcal{C}\}$. The slack variable $\gamma \in \mathbb{R}^m$ and the signal $\omega \in \mathbb{R}^m$ constitute a pair of complementarity variables and $\Gamma(.)$ is some function.

Particularly, if one considers the convex cone $\mathcal{C}$ as the interval $[\ell, u] \in \mathbb{R}_+$, then the complementarity problem given by (1.6c) is transformed to the class of mixed complementarity problems (MCPs) as follows:

$$0 = \mathcal{F}(x, \dot{x}, u, \gamma; t), \tag{1.7a}$$

$$\ell \leq \gamma \leq u \perp \Gamma(\gamma), \tag{1.7b}$$

$$x(t_0) = x_0. \tag{1.7c}$$

where the function $\Gamma(.)$ is multiplied with a negative sign to be a member of $\mathcal{C}^*$ which is $-\mathbb{R}_+$. The complementarity problem given by (1.7b) has been simplified by considering the function $\mathcal{G}$ as $\omega - \gamma = 0$.

Figure 1.9 illustrates solution of the MCP given in (1.7b). This solution is a vector γ such that for each function $\Gamma_i(.), i \in \{1, .., m\}$ one of the following cases holds:

$$\begin{cases} \gamma_i = \ell_i & \text{for} \quad \Gamma_i(\gamma) \geq 0, \\ \ell_i \leq \gamma_i \leq u_i & \text{for} \quad \Gamma_i(\gamma) = 0, \\ \gamma_i = u_i & \text{for} \quad \Gamma_i(\gamma) \leq 0. \end{cases} \tag{1.8}$$

As a result, the complementarity systems given by (1.7) has up to 2^m different modes of operation [31]. It is shown that the class of complementarity systems can be transformed to other classes of hybrid dynamical systems including DIs and the systems with equilibrium constraints [25, 31].

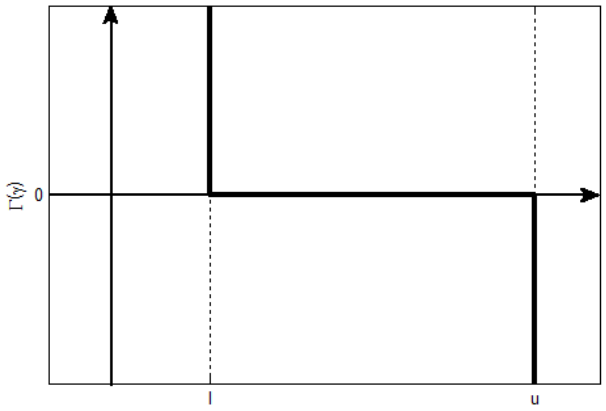


Figure 1.9: Solution of mixed complementarity problems $\mathcal{MCP}(\gamma, \Gamma)$.

1.3.2 Differential inclusions (DIs) of Filippov type

Discontinuous differential states, such as the cycle life variable of batteries, cause the system to be of non-Filippov type [32] which is more challenging for modelling and simulation. Apart from the cycle life, stand-alone dc microgrids can be modeled as differential inclusions (DIs) of Filippov type.

Consider a system given by (1.9) where $\mathcal{F}$ is a non-empty, bounded, and closed set of functions. This piecewise continuous class of differential equations, which is known as differential inclusions [32], presents discontinuous right-hand sides. A differential inclusion given by (1.9) is of Filippov type on an interval $[a, b]$ if differential states remain continuous on that interval.

$$\dot{x} \in \mathcal{F}(\mathsf{x}, \mathsf{u}; t). \tag{1.9}$$

(1.10) indicates a specific representation of the differential inclusions with two modes of operation [25].

$$\dot{x} = \begin{cases} f_-(\mathsf{x}, \mathsf{u}; t) & \text{for} \quad \sigma(\mathsf{x}(t)) < 0, \\ \nu(t) f_-(\mathsf{x}, \mathsf{u}; t) + (1 - \nu(t)) f_+(\mathsf{x}, \mathsf{u}; t) & \text{for} \quad \sigma(\mathsf{x}(t)) = 0, \\ f_+(\mathsf{x}, \mathsf{u}; t) & \text{for} \quad \sigma(\mathsf{x}(t)) > 0. \end{cases} \tag{1.10}$$

where $x(t)$ and $u(t)$ are the states and manipulated control signals, respectively. The switching function $\sigma(x(t))$ determines transitions between two modes of operation while a convex combination of two models is allowed at transition point with $\nu(t) \in [0, 1]$.

1.4 Energy management strategies

During the past decade, a number of researchers conducted their activities towards solving different energy management problems. An energy management strategy specifies the control problem of optimally managing energy flows, with respect to one or more constraint(s), to assure continuous supply of load demand in a power system [2]. The constraints can be economical and technical limitations. The resulting controllers can be assumed as outer controllers that continuously extract the optimum set points which are then applied to some inner controllers such as a pitch angle controller of a wind turbine. To the best knowledge of the author, prior power engineering researches cover a wide range of energy management problems in two main directions: i) grid-connected or stand-alone microgrids; and ii) hybrid or pure electrical vehicles (HEV/EV). However, researches from different backgrounds use different terminologies. In the field of HEV/EV the words energy management system (or strategy) (EMS), power control strategy, and power management system are more common (e.g. [33–36]). On the other hand, researchers in the field of power grids use the words dispatch strategy, energy management system (or strategy), and power management strategy (e.g. [37–39]). The term supervisory control is also common to refer outer level control laws like energy management strategies. In this research the term energy management strategy has been chosen to indicate the employed control law.

An energy management problem consists of a number of sub-problems, including battery management, load and energy resources prediction, system and constraints modelling, fast system simulation as well as appropriate control technologies, to name a few.

There are a number of prior researches focused on modelling uncertainty of renewable energy resources. These models are mostly based on stochastic analysis leading to probability distribution functions (PDFs) of hourly solar irradiance [40–43] and wind speed [44, 45]. These models are essential to simulate energy resources and are applicable to design a reliable renewable site. Moreover, predictions of the available energy resources for some time ahead is vital for probable employing of predictive controllers to manage the energy production and consumption.

Several researchers conducted their research to model different involved components (e.g. [46–51]) to understand dynamical behaviours of a system. Mathematical models of the system make it possible to accurately simulate its behaviour, and also to employ advanced model-based controllers to implement more sophisticated EMSs.

Batteries are one of the most uneconomic components in both microgrids and EVs, principally due to their high cost of replacement which is highly dependent on the charging and discharging methods. Therefore, it is essential to implement a proper battery management strategy as part of any EMS. The researches in [52, 53] have introduced implementations of different battery management strategies as stand-alone systems. Moreover, the works in [21, 35, 54–56] have provided battery management strategies as a part of the developed EMSs.

A number of prior works presented different control techniques to implement EMSs. Some of these strategies are developed by employing conventional control techniques (e.g [21, 37, 50, 57, 58]). Optimal control techniques have become popular recently to solve energy management problems (e.g. [38, 56, 59, 60]), mainly due to handling constraints and the recent advances in relevant numerical solvers.

Through this study, it is shown that the energy management problem of a stand-alone sustainable microgrid can be formulated as an optimal control problem which can be solved by employing non-linear model predictive control (NMPC) technique. Figure 1.10 summarizes the presented optimal EMS in this study from energy, control theories, and involved mathematical concepts perspectives. All applied control theories and involved mathematical concepts are described throughout the next sections.

1.5 Non-linear model predictive control

1.5.1 Open-loop optimal control problems

Optimal control problems (OCPs) make explicit use of the system model in order to find an optimal control law $u^*(.)$ subject to system dynamics and a set of equality and inequality constraints. The term optimal here is defined with respect to certain criteria implying the control objectives. These criteria are specified with a cost functional

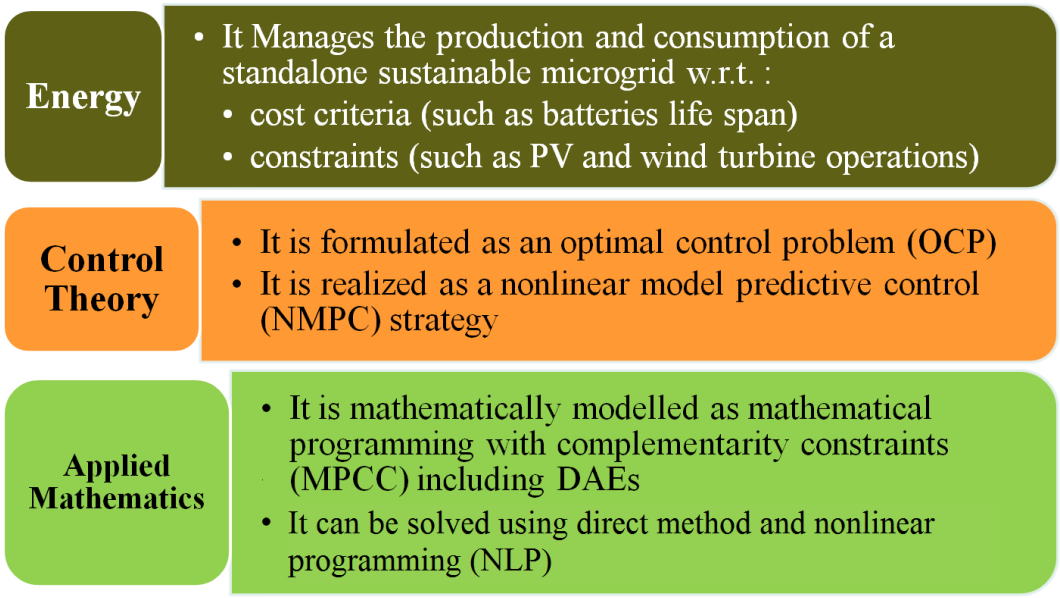


Figure 1.10: An outline of the presented EMS from different perspectives.

J, consisting of Lagrangian term $\mathcal{L}$, also known as the stage cost, that penalizes a distance to the time varying reference trajectory during the prediction period.

It is important to note that an OCP is different from a normal static non-linear optimisation problem in terms of existent dynamics (ordinary or partial differential equations) among its constraints. It makes OCPs member of the class of dynamic optimisation problems. Briefly, it means that the optimum solution of a problem at each sampling instant should optimise trajectories of the system for the infinite future horizon instead of only that specific moment [61].

The structure of a infinite horizon problem, OCP_{∞}, with a cost function J subject to a given non-linear continuous system $\mathcal{F}$ [62], is given in (1.11), where $\mathcal{X}$, $\mathcal{Z}$,and $\mathcal{U}$ denote feasible differential and algebraic states, as well as feasible inputs. From (1.11), it can be seen that there could be initial (in (1.11e)) and final (in (1.11d)) constraints to be maintained by the optimal solution. Moreover, (1.11g) represents boxing constraints on states and control variables.

$$u^*(.) = \underset{\mathsf{u}(.)\in\mathcal{U}^\infty}{\text{arg minimize}} \quad J_\infty(\mathsf{x},\mathsf{z},\mathsf{u};t) := \int_t^\infty \mathcal{L}(\mathsf{x}(\tau),\mathsf{z}(\tau),\mathsf{u}(\tau))d\tau \tag{1.11a}$$

subject to:

$$\mathcal{F}(\mathsf{x}(t),\dot{\mathsf{x}}(\tau),\mathsf{z}(\tau),\mathsf{u}(\tau),\mathsf{v}(\tau)) = 0 \tag{1.11b}$$

$$\mathcal{G}(\mathsf{x}(\tau),\mathsf{z}(\tau),\mathsf{u}(\tau)) \leq 0 \tag{1.11c}$$

$$\mathcal{R}(\mathsf{x}(\infty),\mathsf{z}(\infty)) = 0 \tag{1.11d}$$

$$\mathsf{x}(t) = \mathsf{x}_0, \mathsf{z}(t) = \mathsf{z}_0 \tag{1.11e}$$

$$\forall\tau \in [t, t+\infty] \tag{1.11f}$$

$$\mathsf{x}(\tau) \in \mathcal{X}, \mathsf{z}(\tau) \in \mathcal{Z}, \mathsf{u}(\tau) \in \mathcal{U}. \tag{1.11g}$$

By definition, the minimum value of the cost function with an initial value $\mathsf{x}(t) = \mathsf{x}_0$ and infinite optimisation horizon is called the optimal value function defined as follows:

$$V_\infty(t,\mathsf{x}_0) = J_\infty(\mathsf{x}^*(.),\mathsf{z}^*(.),\mathsf{u}^*(.);t,\mathsf{x}_0) := \underset{\mathsf{u}(.)\in\mathcal{U}^\infty}{\text{minimize}} \quad J_\infty(\mathsf{x},\mathsf{z},\mathsf{u};t,\mathsf{x}_0). \tag{1.12}$$

It is shown that applying an infinite horizon optimal feedback law to the system, the resulting closed-loop system is asymptotically stabilized which can be evaluated with the infinite horizon optimal value function, given in (1.12), as Lyapunov function for the closed loop system [62]. This fact can be analyzed using Bellman's principle of optimality stating that the optimal solution of OCP_∞, given in (1.11), at a specific time instance is part of the optimal solution of such OCP_∞ at the previous time instant. It also stems from Bellman's principle that in nominal cases, with no disturbances available, feasibility at one time instant leads to feasibility and optimality at the next one. It means that if an OCP has a solution at $t = 0$, it will have a feasible solution at $t > 0$ that decreases the value function in (1.12). Choosing (1.12) as the Lyapunov function, its continuity and monotonically decreasing properties at origin establish an asymptotic stability of the closed-loop system.

1.5.2 Non-linear model predictive control

Although one can show the well posedness and stability of the involved OCP_∞ when the infinite horizon optimal control sequence, $u^*(.)$, exists at the time instance $t = 0$, it is not practical, either analytically or numerically, to solve OCP_∞ [62]. In order to be able to numerically solve a non-linear open-loop optimal control problem, it must be formulated as a finite horizon approximation. However, such a finite prediction horizon T, even when there is no disturbances, causes two major issues [63]: First, the shorter finite horizon, the more deviation there would be between the optimal open-loop solution of the finite horizon prediction and real closed-loop trajectories. It is shown that they converge together if the finite horizon approximation problem is solved consecutively with the same cost function over a receding optimisation horizon. Second, there is no guarantee that the optimal solution of the finite horizon approximation stabilizes a closed-loop system. In order to make the resulting closed-loop system stable, the stage cost requires being modified by adding a penalty terms to it, as given in (1.13a), and adding equality or inequality constraints on feasible states, as in (1.13b):

$$J_T(\mathsf{x}, \mathsf{z}, \mathsf{u}; t) := \int_t^{t+T} \mathcal{L}(\mathsf{x}(\tau), \mathsf{z}(\tau), \mathsf{u}(\tau))d\tau + \mathcal{M}(\mathsf{x}(t+T), \mathsf{z}(t+T)), \tag{1.13a}$$

$$\mathsf{x}(t+T) \in \Omega \subseteq \mathcal{X}. \tag{1.13b}$$

While the terminal penalty term $\mathcal{M}$ indicates the cost of state deviations from the desired final values, the term Ω shows terminal region constraint. Both freely chosen $\mathcal{M}$ and Ω are determined suitably in an off-line manner in order to guarantee the stability of the closed-loop system. They can be such that the new cost function in (1.13a) gives an upper approximation of the original infinite horizon cost functional in (1.11a) [63]. In order to find the upper approximation of the infinite horizon cost functional, it can be split into two segments:

$$\begin{aligned} \underset{\mathsf{u}(.)\in\mathcal{U}^\infty}{\text{minimize}} \quad & J_\infty(\mathsf{x}, \mathsf{z}, \mathsf{u}; t, \mathsf{x}_0) = \\ & \underset{\mathsf{u}(.)\in\mathcal{U}^\infty}{\text{minimize}} \quad \Big(\int_t^{t+T} \mathcal{L}(\mathsf{x}(\tau), \mathsf{z}(\tau), \mathsf{u}(\tau))d\tau + \int_{t+T}^{\infty} \mathcal{L}(\mathsf{x}(\tau), \mathsf{z}(\tau), \mathsf{u}(\tau))d\tau \Big). \end{aligned} \tag{1.14}$$

Although it is not possible to find an upper approximation for general non-linear systems, if there is a terminal region around origin within which trajectories of the closed-loop system remain for a time interval of $[t+T, \infty]$, then it is possible to find an upper bound on the second term. This upper approximation of the second term can then be used as the terminal penalty given by term $\mathcal{M}$.

It is important to note that though adding terminal penalty cost is very effective to obtain stability and feasibility of NMPC problems, it is not a necessary condition. There are new advances to stabilize NMPC problems without adding terminal constraints to the original problem (e.g. see [62, 64]).

NMPC, also known as non-linear receding horizon control (RHC), repeatedly solves on-line a finite horizon version of an OCP in order to obtain the optimum control signals at each sampling instant [65]. NMPCs are digital control strategies which are based on discrete finite-horizon OCPs. (1.15) shows the finite horizon discrete version of the OCP given in (1.11). The period of time T in (1.13) is divided into N equal or non-equal length samples with the duration of $h = \frac{T}{N}$ in which the control variables are assumed to be constant [62].

$$u^*(.) = \underset{\mathsf{u}(.)\in\mathcal{U}^N}{\text{arg minimize}} \quad J_N(\mathsf{x}, \mathsf{z}, \mathsf{u}; n) :=$$

$$\sum_{k=n}^{n+N-1} \mathcal{L}(\mathsf{x}(k), \mathsf{z}(k), \mathsf{u}(k)) + \mathcal{M}(\mathsf{x}(n+N), \mathsf{z}(n+N)), \quad (1.15a)$$

subject to:

$$\mathcal{F}(\mathsf{x}(k), \mathsf{x}(k+1), \mathsf{z}(k), \mathsf{u}(k), \mathsf{v}(k)) = 0, \quad (1.15b)$$

$$\mathcal{G}(\mathsf{x}(k), \mathsf{z}(k), \mathsf{u}(k)) \leq 0, \quad (1.15c)$$

$$\mathcal{R}(\mathsf{x}(n+N), \mathsf{z}(n+N)) = 0, \quad (1.15d)$$

$$\mathsf{x}(n) = \mathsf{x}_0, \mathsf{z}(n) = \mathsf{z}_0, \quad (1.15e)$$

$$k \in [n, n+N-1], \quad (1.15f)$$

$$\mathsf{x}(k) \in \mathcal{X}, \mathsf{z}(k) \in \mathcal{Z}, \mathsf{u}(k) \in \mathcal{U}, . \quad (1.15g)$$

Figure 1.11a illustrates a general block diagram of a plant which is controlled by a NMPC. In Figure 1.11a, it can be seen that a NMPC strategy consists of a dynamic

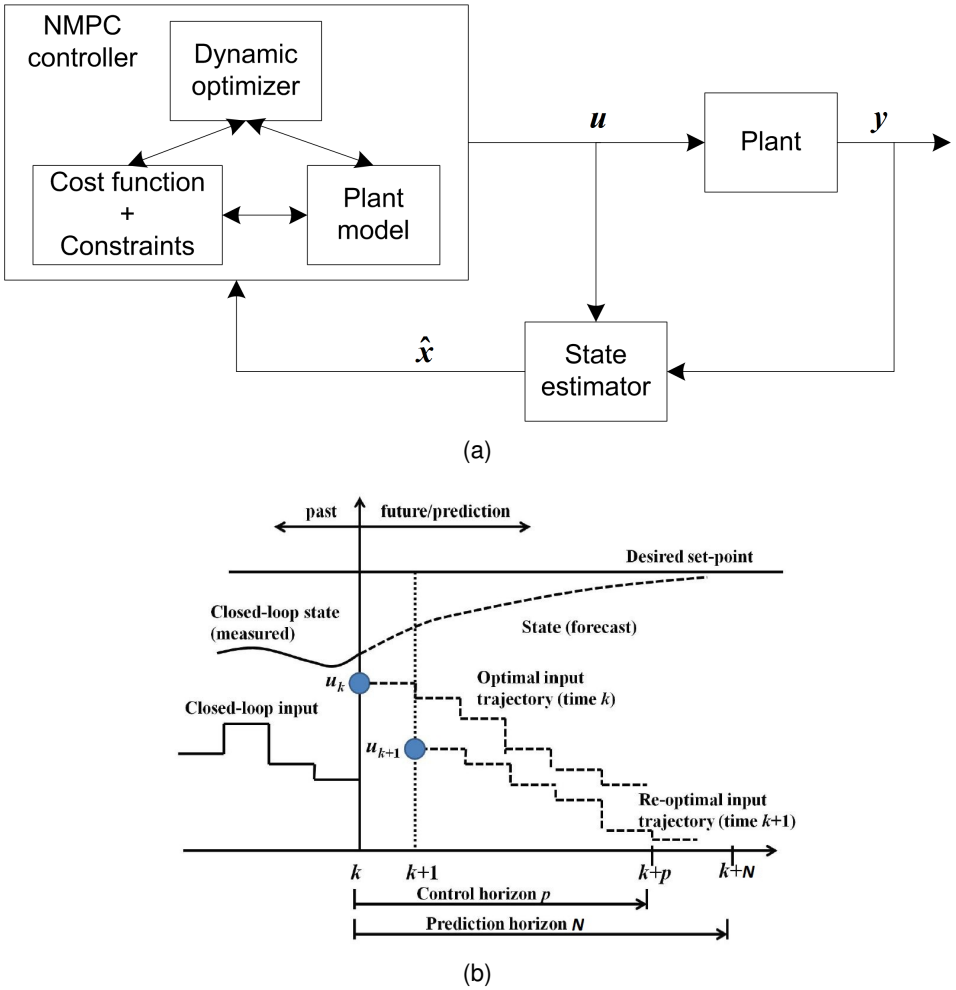


Figure 1.11: General (a) block diagram (courtesy of [63]) and (b) operation of the NMPC.

optimizer, a cost function, as well as a number of constraints. It also relies on, as part of the relevant OCP, a model of the plant to predict its behaviour. The current values of the plant internal states, $\hat{x}$, need to be measured or estimated at k and applied to NMPC as their initial values.

Figure 1.11b shows that, at any sampling instant k, controller calculates the optimum input trajectory $u^*(.)$ for a finite prediction horizon $T = N \times h$ where h is the sampling length. The optimum input trajectory is calculated such that a cost function, e.g. the required time for the closed-loop states to reach desired set-points or the maximum level of overshoots, is minimized. Due to both model and real plant mismatch, as well as measurement disturbances, the predicted and real system behaviours are different and therefore it needs to employ a feed-back mechanism. Controller uses current plant measurements as initial values at k, constructs an OCP and solves it to obtain the optimum open-loop manipulated control signals $u^*(.)$. The first p, where p

usually equals to 1, obtained control signals are then applied to the plant such that the next measurement becomes available at $(k+p)h$ [63]. Using new measurements at $(k+p)h$, controller repeats the procedure with the prediction and control horizons moving forward in order to find a new control law for the next step.

It is important to note that in Figure 1.11b, it can be seen that manipulated closed-loop control signals are discretized as piecewise constant values during the sampling length h.

Unlike conventional causal controllers which use previously computed control laws, NMPC is an acausal technique that computes current control law based on the next N-step ahead predictions of the plant behaviour. Comparing with conventional methods, NMPC, as a non-linear control strategy, presents several advantages.

Due to the existent constraints like actuators limitations, safety limits, and pressure or velocity paths to name but a few, steady-state operation of the plant is normally very close to boundaries of the set of feasible states [65] which is not easy to maintain with conventional control methods; however, NMPC is an inherently non-linear and multivariable strategy that handles constraints and delays [66, 67]. Recently, NMPC became an attractive approach for the control of constrained non-linear processes because of better understanding of its stability and robustness properties and new developed algorithms for dynamic optimisation [66, 68]. Although NMPC is very effective for modern control problems, it suffers from the performance challenge and particularly the computational delay of on-line optimisation which must be quick relative to the plant dynamics in order to achieve desired stability and performance characteristics [66, 68]. Furthermore, there are still several rich research areas relevant to stability, feasibility, and robustness analysis, as well as relevant to appropriate solvers of NMPC strategies for the systems with uncertainties (e.g. [69]) which are the class of stochastic dynamic programming problems. Economic NMPC scheme, which aims at finding optimal non-steady-state points that satisfy all constraints except steady-state ones with a less cost comparing with steady-state points, is also new interesting research area relevant to NMPC (e.g. [70, 71]. Moreover, there are both analytical and numerical challenging problems need to be addressed with regard to applying NMPC strategies for hybrid dynamical systems (e.g. [29]).

1.6 NMPC numerical solvers

In spite of linear model predictive control schemes which lead to solve convex quadratic programming problems with linear constraints, NMPC schemes involve non-linear dynamic programming problems. While there are very effective methods for solving convex problems [72], solving non-linear optimisation problems, in general, are computationally expensive and need to carefully define the optimisation problems and also employ advanced techniques to be able to achieve real-time feasible solutions.

There are three different approaches to solve the NMPCs given by (1.15) [62, 63]: i) dynamic programming method based on Bellman's optimality principle; ii) indirect method based on Pontryagin minimum principle; and finally iii) direct methods, which are used in this study, that convert OCPs into non-linear optimisation problems (NLPs) and use a NLP solver to solve them.

1.6.1 Dynamic programming (DP)

This approach is based on Bellman's principle of optimality which describes a property of the optimal solution of an OCP with horizon N, OCP_N. Figure 1.12 summarizes the Bellman's principle of optimality. From Figure 1.12, it can be seen that regardless of the initial state $x_0 \in \{x_0^0, .., x_0^j\}$ and the first sample of the optimal decision $u^*(0) \in \{u_0^{*0}, .., u_0^{*j}\}$, the remaining optimal decision samples $u^*(1 \leq i \leq N-1)$ construct an optimum solution for the remaining OCP_{N-1} with $x^*(1)$, where $x^*(1)$ is the state value resulting from the first decision $u^*(0)$, as the initial state. In general, it means that in order to solve a discrete OCP_N given by (1.15), one needs to recursively solve $OCP_i, \forall i \in \{1, .., m < N\}$ sub-problems and then combine the solutions to reach an overall solution.

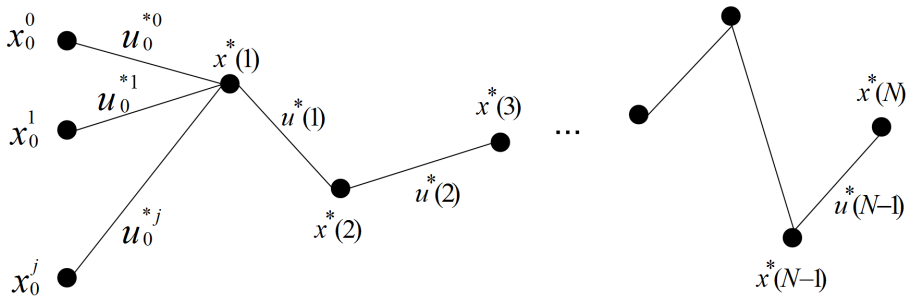


Figure 1.12: A graphical description of Bellman's principle of optimality.

DP is the corresponding discrete-time version, given by (1.16), of the original continuous-time Hamilton-Jacobi-Bellamn (HJB) equation. (1.16) shows Bellman's equation which can be used to iteratively calculate optimal value function from an initial value V_0 with regard to Lagrangian function $\mathcal{L}$ and having the system dynamics $\mathsf{x}(k+1) = f(\mathsf{x}, \mathsf{u}; k)$ [73].

$$V_{K+1}(\mathsf{x}; k) = \underset{\mathsf{u}(.)\in\mathcal{U}^1}{\text{minimize}}\big\{V_K(f(\mathsf{x}, \mathsf{u}; k)) + \mathcal{L}(\mathsf{x}, \mathsf{u}; k)\big\}. \tag{1.16}$$

where $k \in [0, N-1]$ and $K \in [1, N]$.

HJB is a partial differential equation that its solution is the optimal value function of an OCP. Therefore, rather than seeking for the optimal control sequence $u^*(.)$, the problem is approached as finding an optimal value function, given in (1.12), for all initial values which then can be used to calculate corresponding optimal control sequence [73].

Although DP provides a necessary and sufficient condition for optimality leading to find a global optimal solution, a phenomenon named *curse of dimensionality* degrades its performance and restricts its application [73]. The curse of dimensionality states that required computational efforts to solve an OCP by DP grows exponentially with dimension of the states of the system [62] and therefore practically DP is applied to solve small-size problems.

1.6.2 Indirect method

Indirect method employs Pontryagin minimum principle (PMP) as a necessary condition for optimality [61]. PMP states that if there exists an optimal solution for the original OCP, given by (1.15), it must also minimize a *Hamiltonian* function $\mathcal{H}$. The discrete-time Hamiltonian function for an OCP without inequality constraints, i.e. $\mathcal{G}(x, z, u; t) = \varnothing$ in (1.15), is defined as follows:

$$\mathcal{H}(\mathsf{x}, \mathsf{z}, \mathsf{u}, \lambda_{k+1}, \nu_{k+1}; k) := \mathcal{L}(\mathsf{x}, \mathsf{z}, \mathsf{u}; k) + \lambda_{k+1}^T f(\mathsf{x}, \mathsf{z}, \mathsf{u}; k). \tag{1.17}$$

where $k \in [0, N-1]$, f is the explicit form of the system model $\mathcal{F}$ and λ is a vector

of *Lagrange multipliers* that represent the marginal cost of violating equality and inequality constraints.

PMP, as a general case of Euler-Lagrange equation, introduces necessary conditions for the sequence of $u^*(.)$ to be optimal control signals. These conditions are summarized as follows for an OCP without inequality constraints, i.e. $\mathcal{G}(\mathsf{x}, \mathsf{z}, \mathsf{u}; t) = \varnothing$ [61]:

$$\mathcal{H}(\mathsf{x}^*, \mathsf{u}^*, \lambda^*_{k+1}; k) \leq \mathcal{H}(\mathsf{x}^*, \mathsf{u}, \lambda^*_{k+1}; k) \tag{1.18a}$$

$$\mathsf{x}^*(k+1) = \frac{\partial \mathcal{H}}{\partial \lambda_{k+1}}(\mathsf{x}^*, \mathsf{u}^*, \lambda^*_{k+1}; k) \tag{1.18b}$$

$$\lambda^*_k = \frac{\partial \mathcal{H}}{\partial \mathsf{x}}(\mathsf{x}^*, \mathsf{u}^*, \lambda^*_{k+1}; k) \tag{1.18c}$$

$$\lambda_N = \frac{\partial \mathcal{M}}{\partial \mathsf{x}(N)}(\mathsf{x}^*, \mathsf{u}^*; N) \tag{1.18d}$$

$$\mathsf{x}(0) = \mathsf{x}_0 \tag{1.18e}$$

$$\forall \mathsf{u}(k) \in \mathcal{U}, k \in [0, N-1] \tag{1.18f}$$

(1.18a) shows that $u^*(.)$ must minimize Hamiltonian function. In order to minimize Hamiltonian, it is necessary that:

$$\frac{\partial \mathcal{H}}{\partial \mathsf{u}(k)}(\mathsf{x}^*, \mathsf{u}^*, \lambda^*_{k+1}; k) = 0. \tag{1.19}$$

The solution of (1.19) is a parametric local optimum, $u^*(k) = \mu(x^*, \lambda^*_{k+1}; k)$ of PMP and the original OCP in terms of λ^* and x^*. Applying this parametric local optimum to the remaining conditions of PMP, one has a set of differential equations with two boundary constraints. Then this set of differential equations is solved, by using e.g. a shooting method, to extract the optimum values of $x^*(.)$, as optimal trajectories of the original OCP, and $\lambda^*(.)$. It is important to note that if the following condition is satisfied, it provides sufficient condition to guarantee that $u^*(.)$ is the local optimum solution of PMP and so the original OCP:

$$\frac{\partial^2 \mathcal{H}}{\partial^2 \mathsf{u}(k)}(\mathsf{x}^*, \mathsf{u}^*, \lambda^*_{k+1}; k) \succ 0. \tag{1.20}$$

Comparing with DP method and Bellman's optimality principle, PMP is less com-

putational expensive. DP, as a numerical representation of HJB equation, needs a similar computation burden as solving HJB PDEs, while PMP only solves non-linear second-order differential equations [74]. DP computation load increases exponentially in terms of the states dimension comparing to the linear increase of PMP's. However, as mentioned before, the optimal trajectory derived from PMP is not necessarily the global optimum solution and it reduces the computational time to get an optimal solution which could be a local one [74].

1.6.3 Direct methods

Both the necessary and sufficient optimality conditions introduced by DP, as well as the necessary optimality conditions introduced by PMP are increasingly harder to apply to large systems consisting of equality and inequality constraints. For this reason, in order to facilitate the treatment of inequality constraints and singular arcs, NMPC strategies rely on analytic solutions of the state and adjoint equations [25]. Hence, efficient numerical methods are required to apply NMPC strategies to larger systems.

The indirect approach and the resulting PMP necessary conditions can be formulated as a set of DAEs with boundary conditions. The resulting boundary value problem (BVP) can be addressed with different optimise then discretise (O-D) approaches. The O-D based algorithms discretise Euler-Lagrange equations and solve the resulting discretised equation numerically to find the optimal solution.

Due to difficulties associated with solving BVPs and handling inequality constraints, a new discretise and optimise (D-O) approach has been developed. The main aspect of D-O based methods is that the control signals and probably the states are all discretised and the resulting static optimisation problem is then solved using NLP solvers to find optimal solutions. Although this approach requires an accurate level of discretisation for states and control signals, it benefits from direct finding of optimal solution.

There are two groups of strategies employing D-O approach: sequential and simultaneous strategies. Figure 1.13 summarizes a type of the sequential strategy called single shooting. In Figure 1.13, it can be seen that at the inner level the DAE constraints are integrated separately with respect to initial values and control signals

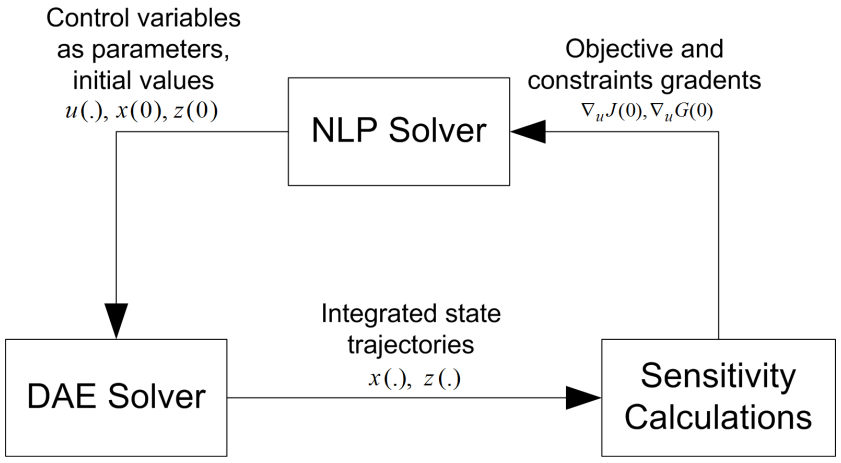


Figure 1.13: An outline of the sequential dynamic optimisation approach [Courtesy of [25]].

given by NLP solver. Calculated state trajectories are used to calculate gradients of the objective and constraints functionals with respect to control variables. Having the gradient values, NLP solver updates control signals and apply them to DAE solver in order to iteratively converge to the optimal solution. In sequential methods, only control variables are discretised as piecewise polynomials and optimisation is performed with respect to the polynomial coefficients. The single shooting strategy is relatively easy to implement which is due to available efficient codes for DAE and NLP solvers. However, it repeats numerical integration of DAE models that degrades its performance. Moreover, it does not guarantee to handle open loop unstable systems [25].

In order to increase the performance of the sequential shooting strategy and also make it usable for unstable systems, single shooting algorithm was modified to a new multiple shooting method [25, 75]. In multiple shooting method, the time domain in sliced into smaller elements and DAE model of the system is integrated separately over each time element. These multiple integrations can be performed in parallel to improve performance of the control strategy. In order to link these time elements to keep the states continuous over time, a number of equality constraints, as initial values of states at each time element, are also added to NLP. These equality constraints ensure a converged solution for unstable systems.

Simultaneous dynamic optimisation strategy, also known as *direct method*, fully discretises both control variables as well as states. This discretization is performed by using collocation on finite elements. Therefore, the existent differential equations of OCP are vanished and the resulting set of algebraic equations and constraints

construct a large non-linear program that can be solved with NLP solvers. Direct method is fully simultaneous and it means that it does not require any inner DAE solver and sensitivity calculator. Moreover, any involved NLP solver benefits from sparsity of the resulting Karush-Kuhn-Tucker (KKT) system leading to improve its performances [25].

In this thesis, direct method is employed to find optimal solutions of developed NMPC strategies. It is chiefly because stand-alone dc microgrids are inherently of the class of hybrid dynamical systems that are difficult to be solved with sequential methods. The next section of this chapter provides a detailed explanation of simultaneous method.

1.6.4 Simultaneous (Collocation) method

Simultaneous method discretises both control variables and states with piecewise polynomial representations are finite elements in time. As a result, dynamic optimisation problems are transformed to non-linear programs which can be solved by NLP solvers with no need of DAE integrators.

In order to develop the collocation method, without loss of generality, the following ODE is considered:

$$\dot{x}(t) = f(x; t). \tag{1.21}$$

According to the collocation method, the ODE given in (1.21) is solved at selected points in time. Although there are a number of polynomial representation, Lagrange interpolation polynomials of degree K is used to approximate state variable $x(t)$ over a single finite element mainly because the polynomial coefficients have the same variable bounds as states profile themselves. Lagrange interpolation polynomials of each time element i is a polynomial of degree K which is constructed by $K + 1$ chosen interpolation points. These interpolation points are chosen for the following time grids:

$$t_{ik} = t_{i-1} + h_i \tau_k, \tag{1.22a}$$

$$\tau_k \in [0, 1], \quad t_{ik} \in [t_{i-1}, t_i]. \tag{1.22b}$$

where $\tau_0 = 0$ and h_i is the length of element i.

Figure 1.14 illustrates a degree $K = 3$ polynomial approximation of the ODE given by (1.21) across an element i of time. Lagrange interpolation polynomials of degree K of each time element i is constructed by $K + 1$ chosen interpolation points as follows:

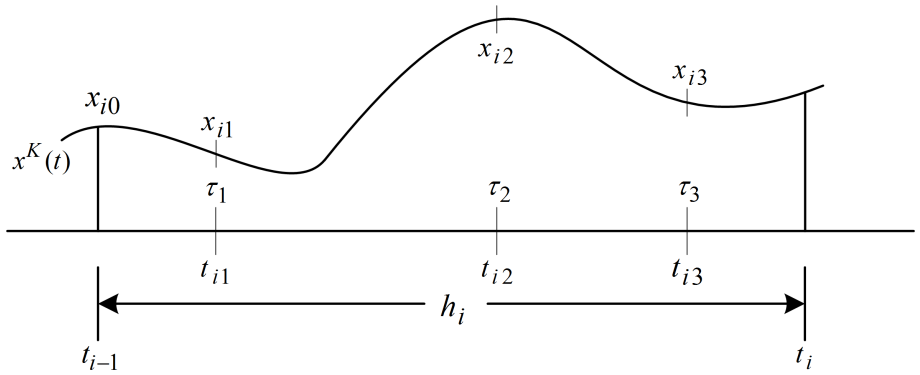


Figure 1.14: The states polynomial approximation across a finite element [Courtesy of [25]].

$$x^K(t_{ik}) = \sum_{j=0}^{K} \ell_j(\tau) x_{ij}, j \in \{0, 1, ..., K-1\}. \tag{1.23a}$$

where $x^K(t_{ij}) = z_{ij}$, $t_{ij} = t_{i-1} + h_i \tau_j$ and:

$$\ell_j(\tau) = \prod_{k=0, \neq j}^{K} \frac{\tau - \tau_k}{\tau_j - \tau_k}. \tag{1.24}$$

Replacing the approximated polynomial into (1.21), the resulting equations at interpolation points τ_k leads to the following collocation equation for each time element i [25]:

$$\begin{aligned} \dot{x}^K(t_{ik}) &= \frac{d}{dt} \sum_{j=0}^{K} \ell_j(\tau_k) x_{ij}, \\ &= \frac{1}{h_i} \sum_{j=0}^{K} \frac{d\ell_j}{d\tau}(\tau_k) x_{ij}, \\ &\approx f(x; t), \\ \therefore \quad \sum_{j=0}^{K} \dot{\ell}_j(\tau_k) x_{ij} &\approx h_i f(x_{ik}; t_{ik}), \quad k = 1, ..., K. \end{aligned} \tag{1.25a}$$

The collocation equation in (1.25) would be a set of algebraic equations if interpolation points τ_k are determined and therefore the resulting algebraic equations can be directly applied into a NLP formulation. With such a representation of differential state variables, control signals and algebraic states can also be discretised by Lagrange interpolation as follows [25]:

$$u(t) = \sum_{j=1}^{K} \overline{\ell}_j(\tau) u_{ij}, \quad z(t) = \sum_{j=1}^{K} \overline{\ell}_j(\tau) z_{ij}. \tag{1.26}$$

where:

$$\overline{\ell}_j(\tau) = \prod_{k=1,\neq j}^{K} \frac{\tau - \tau_k}{\tau_j - \tau_k}. \tag{1.27}$$

The collocation equation given in (1.25) can be extended for DAEs (of index 1) as follows:

$$\sum_{j=0}^{K} \dot{\ell}_j(\tau_k) x_{ij} - h_i f(x_{ik}, z_{ik}, u_{ik}; t_{ik}) = 0, \quad i = 1, .., M, \quad k = 1, ..., K, \tag{1.28a}$$

$$P(x_{ik}, z_{ik}, u_{ik}; t_{ik}) = 0, \quad i = 1, .., M, \quad k = 1, ..., K. \tag{1.28b}$$

Hence, in order to convert an OCP into a non-linear program, interpolation points must be determined such that they provide the most accurate approximation of state variables x and z.

It is proved that the best chosen values for interpolation points τ_k are the roots of a Kth degree orthogonal polynomial, $P_K(\tau)$ [25] such that:

$$\int_0^1 P_j(\tau) P_{j'}(\tau) d\tau = 0, \quad \forall j \neq j', \quad j = 0, ..., K-1, \quad j' = 1, ..., K. \tag{1.29}$$

Shifted Gauss-Legendre and Radau polynomials are among those with the orthogonality property. Table 1.3 provides shifted Gauss-Legendre and Radau roots as collocation points for different degrees K.

For M time elements, (1.30) is required being added to a set of collocation algebraic

equations given in (1.25) in order to enforce continuity of state variable x across different elements boundaries.

$$x_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) x_{ij}, i = 1, ..., M-1 \tag{1.30a}$$

$$x_N = \sum_{j=0}^{K} \ell_j(1) x_{Nj}, \quad x_{1,0} = x_0. \tag{1.30b}$$

Having the collocation discretization of differential and algebraic equations given in (1.25) and (1.28), the relevant NLP formulation corresponding to discretized version of the original OCP_N, given in (1.15), can be constructed as follows:

$$u^*_{nk} = \underset{\mathsf{u}_{nk} \in \mathcal{U}^{N+K}}{\text{arg minimize}} \quad J_N(\mathsf{x}, \mathsf{z}, \mathsf{u}; n) :=$$

$$\sum_{i=n}^{n+N-1} \sum_{j=0}^{K} \mathcal{L}(\mathsf{x}_{ij}, \mathsf{z}_{ij}, \mathsf{u}_{ij}) + \mathcal{M}(\mathsf{x}(n+N), \mathsf{z}(n+N)) \tag{1.31a}$$

subject to:

$$\sum_{j=0}^{K} \dot{\ell}_j(\tau_k) x_{ij} - h_i f(x_{ik}, z_{ik}, u_{ik}; t_{ik}) = 0, \tag{1.31b}$$

$$P(x_{ik}, z_{ik}, u_{ik}; t_{ik}) = 0, \tag{1.31c}$$

$$\mathcal{G}(\mathsf{x}_{ij}, \mathsf{z}_{ij}, \mathsf{u}_{ij}) \leq 0, \tag{1.31d}$$

$$\mathsf{x}_{i+1,0} = \sum_{j=0}^{K} \ell_j(1) x_{ij}, \tag{1.31e}$$

$$\mathsf{x}_{n+N} = \sum_{j=0}^{K} \ell_j(1) x_{(n+N)j}, \quad x_{1,0} = x(t_0), \tag{1.31f}$$

$$\mathcal{R}(\mathsf{x}(n+N), \mathsf{z}(n+N)) = 0, \tag{1.31g}$$

$$i \in \{n+1, ..., n-N+1\}, \quad k \in \{1, ..., N \times K\} \tag{1.31h}$$

$$\mathsf{x}_{ij} \in \mathcal{X}, \mathsf{z}_{ij} \in \mathcal{Z}, \mathsf{u}_{ij} \in \mathcal{U}. \tag{1.31i}$$

In order to find optimal solutions of the original OCP_N, the resulting NLP formulation in (1.31) can be solved by large-scale NLP solvers.

Table 1.3: Shifted Gauss-Legendre and Radau roots as collocation points.

Degree K	Legendre Roots	Radau Roots
1	0.500000	1.000000
2	0.211325	0.333333
	0.788675	1.000000
3	0.112702	0.155051
	0.500000	0.644949
	0.887298	1.000000
4	0.069432	0.088588
	0.330009	0.409467
	0.669991	0.787659
	0.930568	1.000000
5	0.046910	0.057104
	0.230765	0.276843
	0.500000	0.583590
	0.769235	0.860240
	0.953090	1.000000

1.6.5 Non-linear programming (NLP)

Applying direct methods such as simultaneous collocation approach to solve NMPC problems, converts the original problems to large-scale NLP problems with both equality and inequality constraints [68] as follows:

$$\underset{z\in\Re^n}{\text{minimize}} \quad F(z) \tag{1.32a}$$

$$\text{subject to:} \quad P(z) = 0 \tag{1.32b}$$

$$G(z) \geq 0. \tag{1.32c}$$

where the vector of functionals P and G are the equality and inequality constraints of the problem, respectively.

In order to solve the resulting large-scale NLP problems with both equality and inequality constraints, there are at least three approaches [25]: sequential quadratic programming (SQP)-type methods, interior-point methods, and nested projection methods. There are some prior researches that compared these different methods for NMPC applications (e.g. see [76]). During the past decade an open-source

version of the interior-point method, called IPOPT, has been developed to solve large-scale NLPs [68]. IPOPT employs a logarithmic barrier method to construct a relaxed version of the original NLP. IPOPT is the chosen NLP solver in this study, chiefly because of its excellent convergence properties and superior performance for the dynamic optimisation problems [68]. Furthermore, IPOPT is available as an open-source packages. A summary of the interior-point method and particularly logarithmic barrier method is provided and for further details, the reader is referred to [72] and [25].

Interior-point method (IPM)

The Interior-point method (IPM), comparing with the active set based strategies like sequential quadratic programming (SQP), is largely insensitive to the number of active constraints [76]. Moreover, since it iteratively solves a sparse linear system, it exploits the sparsity structure of the original NMPC problem [76]. The fundamental idea of IPM is to modify the NLP problem given in Eq. (1.32) under consideration that all constraints become equality constraints. There are two main approaches to achieve this aim, namely, *homotopy*, and *barrier* approaches. Here, a summary of both approaches has been provided and the reader is referred to [72] and [25] for further details.

According to the homotopy approach all inequality constraints are converted to equality ones using a vector of slack variables s so then (1.32) is transformed to (1.33) as follows:

$$\underset{z\in\Re^n, s\in\Re^{n_s}}{\text{minimize}} \quad F(z) \tag{1.33a}$$

$$\text{subject to:} \quad P(z) = 0 \tag{1.33b}$$

$$G(z) - S = 0. \tag{1.33c}$$

where n_s denotes dimension of slack variables equals to the number of inequality constraints and $S = diag(s)$. Now applying Karush-Kuhn-Tucker (KKT) theorem a system of differential and algebraic equations are constructed as a necessary conditions. KKT theorem states that for an optimal point z^* of (1.33) there exist

multipliers λ^* and ν^* and slack variables s^* such that a system of equations:

$$\nabla_z \mathscr{L}(z^*, \lambda^*, \nu^*) = 0 \tag{1.34a}$$

$$S\nu - \nu e = 0 \tag{1.34b}$$

$$P(z^*) = 0 \tag{1.34c}$$

$$G(z^*) - S = 0 \tag{1.34d}$$

$$s^* \geq 0. \tag{1.34e}$$

holds where $e = [1, 1, ..., 1]^T$ and υ is an iteratively vanishing perturbation parameter, i.e. $\lim_{j \longrightarrow \infty} \upsilon_j = 0$, in order to enforce the solution to stay away from boundaries, and $\mathscr{L}$ is Lagrangian function of (1.33) as follows:

$$\mathscr{L}(z, \lambda, \nu) = F(z) - P(z)^T \lambda - G(z)^T \nu. \tag{1.35}$$

Different methods such as linesearch or trust-region can be employed to solve the resulting (1.34) iteratively to find the set of $(z_{j+1}, \lambda_{j+1}, \nu_{j+1}, s_{j+1})$ with an initial values $(z_j, \lambda_j, \nu_j, s_j)$.

On the other hand, according to barrier approach, one uses a *log*-barrier function and adds it as a penalty to the cost function F to construct a relaxed problem with barrier parameters $\{\mu_l\}$ as follows:

$$\underset{z \in \Re^n, s \in \Re^{ns}}{\text{minimize}} \qquad F(z) - \mu_l \sum_{i=1}^{n_G} ln(G(z)) \tag{1.36a}$$

$$\text{subject to:} \quad P(z) = 0 \tag{1.36b}$$

$$G(z) > 0. \tag{1.36c}$$

where n_x represents number of inequality constraints G and integer l is a sequence counter such that $\lim_{j \longrightarrow \infty} \mu_l = 0$. It is important to note that since the *log*-barrier term becomes unbounded at $G(z) = 0$, the inequality constraints $G(z)$ is strictly positive. As the barrier parameters $\{\mu_l\}$ decreases to zero, the solution of the relaxed problem in (1.36) converges to a local solution of the original NLP problem given in

(1.32). Hence, in order to solve the original NLP, a sequence of barrier problems is constructed with monotonically decreasing sequence of barrier parameters $\{\mu_l\}$ that converges to the solution of the original NLP. Similar to *homotopy* approach, a Newton based method can be applied to iteratively solve the derived KKT necessary conditions, leading to solution of the following sparse linear system at each iteration j to find the set of $(z_{j+1} = z_j + \Delta z, \lambda_{j+1} = \lambda_j + \Delta\lambda, \nu_{j+1} = \nu_j + \Delta\nu)$:

$$\begin{bmatrix} W_j & A_j & -I \\ A_j^T & 0 & 0 \\ Z_j & 0 & X_j \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta\lambda \\ \Delta\nu \end{bmatrix} = - \begin{bmatrix} \nabla F(z_j) + A_j\lambda_j - \nu_j \\ P(z_j) \\ Z_j N_j e - \mu_l e \end{bmatrix} \tag{1.37}$$

where $e = [1, 1, ..., 1]^T$ and we define $Z = diag(z)$, $N = diag(\nu)$, the Hessian $W_j = \nabla_{zz}\mathscr{L}(z_j, \lambda_j, \nu_j)$, and $A_j = \nabla P(z)$.

1.6.6 Automatic differentiation (AD)

The performance of NLP solvers, and hence NMPC strategies, strongly depends on accuracy and speed of the employed Jacobian, and Hessian matrix calculation algorithms. Therefore, differentiation is a fundamental problem of numerical NLP solvers and is one the main bottlenecks for their performance and accuracy.

Automatic differentiation (AD) is a state-of-the-art technique to compute fast and accurate derivatives of any degree. Although it is not neither numerical nor symbolic differentiation method, it can be seen as a numerical method that also gets benefits from the symbolic differentiation idea; however, it is not limited by neither complexity of equations nor derivative degree. It does not endure truncation errors, annoying other numerical methods.

AD relies on the fact that all computer programs execute a finite set of elementary operations, defined by the programming languages. AD evaluates derivative of a function specified by a computer program, consisting of a finite set of elementary operations. The partial derivative of elementary operations are known and therefore AD generates another computer program to calculate the overall derivative of such a function by assigning intermediate variables to each elementary operation and using the chain rule. There are two basic modes of AD: *forward mode* and *reverse mode*.

By forward mod, the derivatives are propagated throughout computations from basic variables to the top level of function. On the other hand, in reverse mode, derivatives are propagated from the top level of function down to variables.

Table 1.4 summarizes operation of AD forward mode to differentiate a sample function $f(x) = (x + x^2)^2$. Numerical value of the evaluated derivative is calculated by *seeding* the variable x and its derivative with the values 2 and 1, respectively.

Table 1.5, on the other hand, summarizes operation of AD reverse mode to differentiate the same function. In this mode, $\overline{w}$ represents derivative of the output y with respect to w and therefor $\overline{y} = 1$. For the reverse mode, numerical value of the evaluated derivative is calculated by seeding variable x.

Table 1.4: Automatic differentiation of a sample function $y = f(x) = (x + x^2)^2$ in the forward mode.

Elementary operations	Forward derivatives	Intermediate variables Values	Derivatives Values
$w_1 = x = 2$	$\dot{w_1} = \dot{x} = 1$		
$w_2 = x^2 = w_1^2$	$\dot{w_2} = 2w_1\dot{w_1}$	4	4
$w_3 = x + x^2 = w_1 + w_2$	$\dot{w_3} = \dot{w_1} + \dot{w_2}$	6	5
$w_4 = (x + x^2)^2 = w_3^2$	$\dot{w_4} = 2w_3\dot{w_3}$	36	60

Table 1.5: Automatic differentiation of a sample function $y = (x + x^2)^2$ in the reverse mode.

Elementary operations	Reverse derivatives	Intermediate variables Values	Derivatives Values
$w_4 = (x + x^2)^2 = w_3^2$	$\overline{w_4}$	36	1
$w_3 = x + x^2 = w_1 + w_2$	$\overline{w_3} = 2w_3\overline{w_4}$	6	12
$w_2 = x^2 = w_1^2$	$\overline{w_2} = \overline{w_3}$	4	12
$w_1 = x = 2$	$\overline{w_1} = (1 + 2w_1)\overline{w_3}$	60	

Complete Jacobians and Hessian can be computed by repeating AD procedure. AD considerably saves the evaluation costs comparing to numerical methods [77]. Jacobian is evaluated with a cost proportional to the number of independent variables, in forward mode, and dimension of the function, in reverse mode.

For further details on AD, the reader is referred to [78–80].

Computer algebra system with automatic differentiation implementation (CasADi)

There are a number of AD tools for C/C++, FORTRAN, and MATLAB functions. Among available tools, computer algebra system with automatic differentiation im-

plementation (CasADi) is employed in this research. CasADi is an open-source integrated framework for non-linear programming in C/C++ or Python, and consists of an implementation of AD as well as interfaces to NLP and DAE solvers [81]. Moreover, it is a computer algebra system (CAS) tool to implement different direct discretization algorithms, including single and multiple shooting and collocation methods.

Providing AD, CAS, and interfaces to DAE and NLP solvers, CasADi is a flexible tool to address optimal control problems and, in particular, NMPC strategies. Instead of providing the end user with a black box OCP solver, as it is in other existing OCP tools such as ACADO [82] and MUSCOD-II [83], CasADi is a low-level developing tool and allows advanced users to implement their own NMPC strategy.

For further details on CasADi, the reader is referred to [77, 81].

1.6.7 NMPC strategies for hybrid dynamical systems

Different control theories, including NMPC, were originally developed for continuous systems; however, there is a great interest in expanding such theories for hybrid dynamical systems. Although DAEs are effective to model continuous systems, discontinuities available in hybrid systems embed new details to the model which are described with different techniques (see Section 1.3). NMPC is independent of the type of underlying system, though the system model magnificently affects the computational complexity, stability, and robustness issues. In particular, different tools are required to solve NMPC problems with different types of prediction model.

The author in [25] stated that NMPC strategies to control differential inclusions of Filippov type, which is targeted in this study, can be solved by employing collocation method. Applying collocation method to a NMPC problem, such a problem is transfered into a mathematical programming with complementarity constraints (MPCC). However, in order to transform a NMPC problem to a MPCC, it should be of the complementarity class of hybrid dynamical systems [31]. Employing a penalization method, the resulting MPCC can be transformed into a non-linear programming (NLP) problem, instead of mixed integer non-linear programming (MINLP), and can be solved by general purpose NLP solvers.

Mathematical programming with complementarity constraint (MPCC)

As described in Section 1.3, a complementarity constraint is a relationship which is defined between a number of variables in which at least one, or more, of variables is at its bounds. A NLP with embedding such a complementarity constraint is inherently nonconvex. Moreover, such complementarity constraints introduce linear dependence of constraints, which makes NLP harder to be solved [84, 85].

MPCCs are a specific form of the more general bilevel optimisation problems, called as mathematical programming with equilibrium constraints (MPECs), with the following simplified form:

$$\underset{x\in\Re^{n_c}}{\text{minimize}} \quad F(x,y) \tag{1.38a}$$

$$\text{subject to:} \quad P(x,y)=0 \tag{1.38b}$$

$$G(x,y)\leq 0 \tag{1.38c}$$

$$y=\underset{\hat{y}\in\Re^{n_c}}{\text{arg minimize}}\{x^T\hat{y}:\hat{y}\geq 0\}. \tag{1.38d}$$

(1.38d) indicates the lower level optimisation problem within the outer problem. The value of y is zero for $\forall x>0$ and it is greater than zero for $\forall x\leq 0$. The lower level optimisation constraint can be converted into a complementarity constraint on the outer problem. As a result, one gets a MPCC as follows:

$$\underset{x,y,z}{\text{minimize}} \quad F(x,y,z) \tag{1.39a}$$

$$\text{subject to:} \quad P(x,y,z)=0 \tag{1.39b}$$

$$G(x,y,z)\leq 0 \tag{1.39c}$$

$$0\leq x\perp y\geq 0. \tag{1.39d}$$

where $x\in\Re^{n_c}$ and $y\in\Re^{n_c}$ are the complementing variables and $z\in\Re^{n-2n_c}$ is the remaining variable.

A MPCC of the form of (1.39) needs to be reformulated in order to be solved with general NLP solvers [25]. According to the necessary KKT conditions for optimality,

a NLP problem has a feasible solution if constraints qualifications hold [25, 84, 85]; however, these qualifications are violated for MPCCs.

Two types of MPCC reformulation techniques, namely, *regularized* and *penalized* formulations, have been analyzed in prior researches [25]. According to the regularized formulation, the complementarity constraint given in (1.39d) is relaxed with a positive relaxation parameter ϵ. The solution of MPCC is obtained by solving a series of relaxed solutions as ϵ approaches zero. On the other hand, in penalized formulation, complementarity constraints are moved to the objective function and the resulting problem is solved for a particular large enough penalty parameter ρ. The MPCC problem in (1.39) is regularized and penalized, respectively, in (1.40) and (1.41). It is important to note that (1.40) shows a regularized formulation of the original MPCC problem in which the complementarity constraint is relaxed with the equality in (1.40e). However, it can be relaxed in different ways [25].

$$\underset{x,y,z}{\text{minimize}} \quad F(x,y,z) \tag{1.40a}$$

$$\text{subject to:} \quad P(x,y,z) = 0 \tag{1.40b}$$

$$G(x,y,z) \leq 0 \tag{1.40c}$$

$$x, y \geq 0 \tag{1.40d}$$

$$x_i y_i \leq \epsilon, \quad i = 1, ..., n_c. \tag{1.40e}$$

$$\underset{x,y,z}{\text{minimize}} \quad F(x,y,z) + \rho x^T y \tag{1.41a}$$

$$\text{subject to:} \quad P(x,y,z) = 0 \tag{1.41b}$$

$$G(x,y,z) \leq 0 \tag{1.41c}$$

$$x, y \geq 0. \tag{1.41d}$$

For further details on MPCC and NMPC for hybrid dynamical systems, the reader is refer to [25, 31, 84–87].

1.7 Research objectives and contributions

Energy management strategies are essential elements of stand-alone green microgrids for the power sharing and voltage regulation purposes. Conventionally, energy management strategies employ decoupled maximum power point tracking (MPPT) algorithms and rely on batteries to absorb excess energy. However, in order to realize constant current-constant voltage (IU) charging regime and increase the life span of batteries, energy management strategies would be more flexible and equipped with a power curtailment feature. Such a non-linear multivarible control problem with operational constraints requires an advanced control strategy. In order to analyze different operations of microgrids and to develop model-based control strategies, microgrids need to be modelled mathematically. However, microgrids belong to the class of hybrid dynamical systems of non-Filippov type, because of bimodal operation of batteries and their discontinuous differential states, which is not practical for controllers design.

Moreover, a microgrid consists of a number of physical components which must be mathematically modelled. PV modules, wind turbines, generators, and batteries are physical components with parameters which must be identified appropriately. For instance, an accurate and fast PV model identification and maximum power point (MPP) estimation method are becoming essential to reduce the PV power losses effectively.

This study targets the following objectives in order to address the above mentioned issues:

- Providing a mathematical model of stand-alone sustainable microgrids for the long-term simulation purpose as well as for developing control strategies.
- Introducing a novel PV model identification algorithm to identify parameters of the equivalent electrical circuit of a PV module.
- Conducting a study on stochastic analysis of renewable energy resources, i.e. solar irradiance and wind speed.
- Providing a framework to conduct long-term simulation of a sample stand-alone dc microgrid to be able to predict its operation over a long time and under

energy resources uncertainties.

- Developing non-linear multivariable control strategies for stand-alone dc microgrids to achieve five control objectives: i) supplying variable dc loads; ii) regulating the dc bus voltage level; iii) charging batteries as close to IU regime as possible; iv) proportional power sharing between generators; v) tracking MPPs of wind and PV branches during normal operations.

Chapter 2

Acausal Modelling of Stand-alone dc Microgrids as Complementarity Systems

Due to bimodal operation as well as existent discontinuous differential states, batteries and therefore microgrids belong to the class of hybrid dynamical systems of non-Filippov type. In order to develop a mathematical model applicable to NMPC strategies, stand-alone dc microgrids are approximated as complementarity systems of Filippov type. The presented model is also used to develop a Modelica model for the long-term simulation purposes.

2.1 Introduction

Microgrids are the building blocks of modern power grids. As explained in Chapter 1, due to some challenges that ac microgrids face with [6, 3], dc microgrids get more popular particularly for stand-alone applications [1, 88]. There are new interests to employ NMPC technique [63, 62] to develop coordinated multivariable control strategies for stand-alone microgrids (e.g. [89, 90]). However, such control strategies require using a mathematical model of microgrids to predict their behavior during the prediction horizon. Moreover, in smart grid applications, such a model is needed to simulate the microgrids behavior for at least one day ahead [91]. There are three major challenges regarding developing a mathematical model for microgrids: i) the

algebraic constraints presented by PV module, wind turbine, and battery bank; ii) battery bank as a sub-system with two modes of operation, namely, charging and discharging; and iii) cycle life of battery bank as a discontinuous differential state.

The available algebraic constraints and bimodal operation of battery bank cause the mathematical model of stand-alone dc microgrids to be a set of hybrid differential algebraic equations (hybrid DAEs) [22]. Therefore, stand-alone dc microgrids are represented by an acausal model for the control and simulation purposes. Unlike causal approach, which requires the system to be decomposed into a chain of causal interacting blocks consisting of only ordinary differential equations (ODEs), acausal modeling is a declarative approach in which individual parts of the model are described as hybrid DAEs [22]. Acausal modeling, as the mathematical representation of the system, is an effective way to model and simulate complex systems and such a model is also applicable to develop NMPC strategies. Modelica [23], as an object-oriented and equation-based language to describe complex systems, provides the capability to acausally model the class of hybrid dynamical systems [22].

Cycle life of batteries, as discontinuous differential states, cause the system to be of non-Filippov type [32] which is more challenging. Cycle life is defined as the total number of complete charging/discharging cycles a battery can undergo before its capacity falls down below 80% of its nominal value. Apart from the cycle life, stand-alone dc microgrids can be modeled as differential inclusions (DIs) of Filippov type. The work in [25] argues that NMPC strategies to control the hybrid systems of Filippov type may be solved by employing a variable element length version of the collocation method. Applying this flexible version of collocation method transfers such NMPC problems into a mathematical programming with MPCC. Then, employing a penalization method, the resulting MPCC can be transformed into a NLP problem which can be solved by general purpose NLP solvers. However, in order to transform NMPC problem to a MPCC, such problem should be of the complementarity class of hybrid dynamical systems [31].

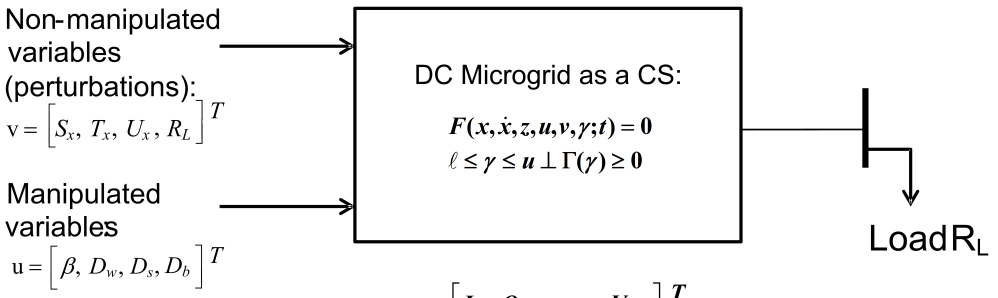
The main aspect of this chapter is to provide a mathematical model of stand-alone dc microgrids applicable to NMPC strategies and long-term simulation purposes. For this aim stand-alone dc microgrids are modeled as CSs including differential and al-

gebraic constraints as well as MCPs. The discontinuous cycle life differential state is reformulated with a continuous approximation and such a reformulation transforms the proposed dc microgrid model to be of Filippov type. Moreover, bimodal operation of battery bank is also modeled as separate complementarity constraints. The developed model is then used in two different applications: i) in Chapter 5, an equivalent model in the form of DIs, is employed to simulate stand-alone dc microgrids; and ii) in Chapter 6, a number of NMPC strategies based on the presented model are developed to track load demands and regulate dc bus voltage despite substantial generation fluctuations.

The remainder of this chapter is organized as follows. Section 2.2 deals with the modeling of a stand-alone dc microgrid as a CS followed by Section 2.3 that presents and discusses the developed Modelica model of stand-alone dc microgrids which is applicable for the long-term simulation purpose.

2.2 Modelling of stand-alone dc microgrids as complementarity systems

Figure 2.1 summarizes the developed model for the stand-alone dc microgrid topology given in Figure 1.1. It can be seen that the dc microgrid is modeled as a CS including mixed complementarity problems with slack variables γ. Such a CS consists of differential and algebraic states, i.e. x and z, and manipulated and non-manipulated control variables, namely, u and v, which are detailed later throughout the next sub-sections.



$$x = \left[I_f,\ Q_{act},\ \omega_r,\ V_{exp} \right]^T$$

$$z = \left[I_{pv}, V_{pv},\ I_{pvdc},\ I_{bstack},\ V_{bstack},\ I_{batdc},\ I_{wt}, V_{wt},\ I_{wtdc},\ T_e,\ T_m,\ \lambda,\ C_p,\ SOC,\ I_{load},\ V_{dc} \right]^T$$

$$\gamma = \left[\gamma_1,\ \gamma_2 \right]^T$$

Figure 2.1: DC microgrids as complementarity systems.

The following notations are used to model the stand-alone dc microgrid depicted in Figure 1.1 as a CS given by (1.7):

$$\mathcal{F}(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) = \begin{bmatrix} f_1(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \\ f_2(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \\ .. \\ f_{25}(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \end{bmatrix} = 0, \tag{2.1a}$$

$$\mathcal{MCP}(\gamma, \Gamma) = \begin{bmatrix} \mathcal{MCP}_1(\gamma_1, \Gamma_1) \\ \mathcal{MCP}_2(\gamma_2, \Gamma_2) \end{bmatrix}. \tag{2.1b}$$

where $\mathcal{F}$ is a set of implicit differential and algebraic functionals f_i for $i \in \{1, 2, .., 25\}$ that models dc microgrids behaviors. All the equations throughput the chapter which are not indicated as implicit functions $\{f_i\}$ are used to calculate intermediate variables. $\mathcal{MCP}$ is a vector of two MCPs as in (1.7b) that models different modes of operation and any transitions between them.

The following sub-sections present all differential and algebraic equations that model battery, solar, and wind branches. Connecting all these branches together, there are two power balance constraints given by (2.2) indicating that the sum of the generation and the consumption powers should always be zero.

$$f_1 = V_{dc}\Big(I_{pvdc} + I_{wtdc} + I_{batdc} - I_{load}\Big), \tag{2.2a}$$

$$f_2 = V_{dc} - I_{load}R_L. \tag{2.2b}$$

2.2.1 Solar branch

This section presents a mathematical model of solar branch, consisting of a PV array that is connected to the dc bus through a boost-type dc-dc converter. The presented model is based on the equivalent electrical circuit of a PV module [46, 92]. The average model of converters can be replaced with the steady-state equations

in CCM as follows:

$$f_3 = V_{pv} - (1 - D_s)V_{dc}, \tag{2.3a}$$

$$f_4 = I_{pvdc} - (1 - D_s)I_{pv}. \tag{2.3b}$$

A PV array consists of, respectively, N_{pvs} and N_{pvp} PV modules in series and parallel. Given photocurrent, I_{ph}, and diode reverse saturation current, I_0, in (2.4b) and (2.4c), one can model a PV array with the transcendental characteristic equation given by (2.4a) [46].

$$f_5 = I_{pv} - I_{ph} + \frac{V_{pv} + \frac{N_{pvs}}{N_{pvp}} R_s I_{pv}}{\frac{N_{pvs}}{N_{pvp}} R_{sh}} + I_0 \left\{ exp(\frac{V_{pv} + \frac{N_{pvs}}{N_{pvp}} R_s I_{pv}}{n_d N_s} \frac{q \times N_{pvs}}{K T_c}) - 1 \right\}, \tag{2.4a}$$

$$f_6 = I_{ph} - N_{pvp} \times \left(\frac{R_s + R_{sh}}{R_{sh}} I_{sc,stc} + k_I \left(T_c - T_{c,stc} \right) \right) \frac{S}{S_{stc}}, \tag{2.4b}$$

$$f_7 = I_0 - N_{pvp} \times \frac{I_{sc,stc} + k_I \left(T_c - T_{c,stc} \right)}{\exp(\frac{V_{oc,stc} + k_V \left(T_c - T_{c,stc} \right)}{n_d N_s} \frac{q}{K T_c}) - 1}. \tag{2.4c}$$

where R_s and R_{sh}, respectively, are series and parallel equivalent resistors of PV module and all other parameters are as follows:

q	Electron charge (1.60218×10^{-19})
N_s	The number of PV cells in PV module (-)
K	Boltzman constant (1.38066×10^{-23})
k_I	The short-circuit current temperature coefficient ($A/^\circ C$)
T_c	The current PV cell temperature (K)
$I_{sc,stc}$	Short-circuit current at STC (A)
S	The current solar irradiance (W/m^2)
k_V	Open-circuit voltage temperature coefficient ($V/^\circ C$)
S_{stc}	Insolation level (W/m^2)
$T_{c,stc}$	STC cell temperature (K)
$V_{oc,stc}$	Open-circuit voltage at STC (V)

2.2.2 Battery branch

A model to mathematically represent lead-acid batteries is introduced in [20]. This model can be generalized as a CS by introducing the following MCP:

$$\mathcal{MCP}_1 : 0 \leq \gamma_1 \leq 1 \quad \perp \quad \Gamma_1(\gamma_1) = I_{bstack}, \tag{2.5}$$

where I_{bstack} is the current of a battery stack. Regarding the solution of a MCP expressed by (1.8), γ_1 is a slack variable which indicates if the battery bank is in charging ($\gamma_1 = 1$) or discharging ($\gamma_1 = 0$) mode of operation. It is important to note that according to the notation throughout this chapter, I_{bstack} is negative during charging mode of operation.

Therefore, a lead-acid battery bank, consisting of N_{batp} battery stacks in parallel while each stack consisting of N_{bats} batteries in series, can be modeled as the DAEs coupled with the complementarity constraint given by (2.5). Notice that at each moment just one of the segments of equations is active in terms of the value of γ_1:

$$f_8 = \frac{V_{bstack}}{N_{bats}} + \boldsymbol{\gamma_1}\big\{R_{bat}\frac{I_{bstack}}{N_{batp}} - V_0 + \frac{P_1 C_{max}}{C_{max} - Q_{act}}Q_{act} + \frac{P_1 C_{max}}{0.1C_{max} + Q_{act}}I_f - V_{exp}\big\} + \\ \boldsymbol{(1-\gamma_1)}\big\{R_{bat}\frac{I_{bstack}}{N_{batp}} - V_0 + \frac{P_1 C_{max}}{C_{max} - Q_{act}}Q_{act} + \frac{P_1 C_{max}}{C_{max} - Q_{act}}I_f - V_{exp}\big\}. \tag{2.6a}$$

$$f_9 = \frac{\mathrm{d}Q_{act}}{\mathrm{d}t}(t) - \frac{1}{3600}\frac{I_{bstack}(t)}{N_{batp}}, \tag{2.6b}$$

$$f_{10} = \frac{\mathrm{d}I_f}{\mathrm{d}t}(t) + \frac{1}{T_s}(I_f - \frac{I_{bstack}}{N_{batp}}), \tag{2.6c}$$

$$f_{11} = \frac{\mathrm{d}V_{exp}}{\mathrm{d}t}(t) - \boldsymbol{\gamma_1}\big\{\frac{P_2}{3600}\left|\frac{I_{bstack}}{N_{bats}}\right|(P_3 - V_{exp})\big\} + \boldsymbol{(1-\gamma_1)}\big\{\frac{P_2}{3600}\left|\frac{I_{bstack}}{N_{bats}}\right|V_{exp}\big\}, \tag{2.6d}$$

$$f_{12} = V_{bstack} - \frac{V_{dc}}{1 - D_b}, \tag{2.6e}$$

$$f_{13} = I_{bstack} - (1 - D_b)I_{batdc}, \tag{2.6f}$$

$$f_{14} = SOC - \left\{1 - \frac{Q_{act}}{C_{max}}\right\}. \tag{2.6g}$$

V_{bstack}, I_{bstack} and SOC, respectively, are voltage, current, and state of charge of battery stack. Parameters $P_1 - P_3$ are experimental and require to be identified for each type of battery. The C_{max} is the maximum amount of the battery capacity, R_{bat} is internal resistor of battery, Q_{act} is the actual battery capacity, and V_0 is the

battery constant voltage. Moreover, I_f is the filtered value of the battery current with a time constant of T_s [20]. The bi-directional converter with duty cycle D_b for the boost switching power pole is modeled in steady-state continuous conduction mode (CCM) [13].

The cycle life of a battery bank, as a cost measure of its delivered energy [93], mainly depends on two factors: the number of charging/discharging cycles and depth of each discharge. Figure 2.2a illustrates the normalized cycle life as a discontinuous differential state being increased by the reverse of equivalent full cycle (EFC) at the moment of each mode switching from discharging to charging. EFC indicates the number of cycles of a battery at any depth of discharge (DOD) which is given by a curve for different values of DOD [93].

In order to model the cycle life as a continuous differential state, and therefore convert the system to the class of Filippov type, a new continuous state, t_{chg}, is defined as in Figure 2.2b which shows duration of charging cycle. The t_{chg} starts from zero and monotonically increases during charging cycle. Once battery switches to discharging mode, t_{chg} sharply drops down to zero. Such a differential state can be defined using the following complementarity problem:

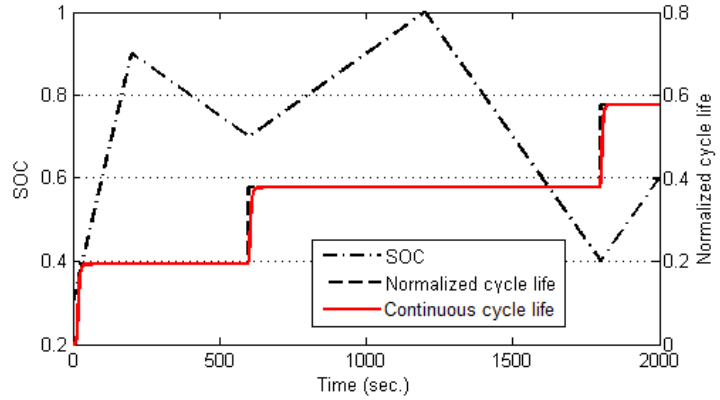
$$\mathcal{MCP}_2 : 0 \leq \gamma_2 \leq 1 \quad \perp \quad \Gamma_2(\gamma_2) = -t_{chg}, \tag{2.7}$$

where γ_2 is a slack variable shows if t_{chg} is positive ($\gamma_2 = 1$) or not ($\gamma_2 = 0$).

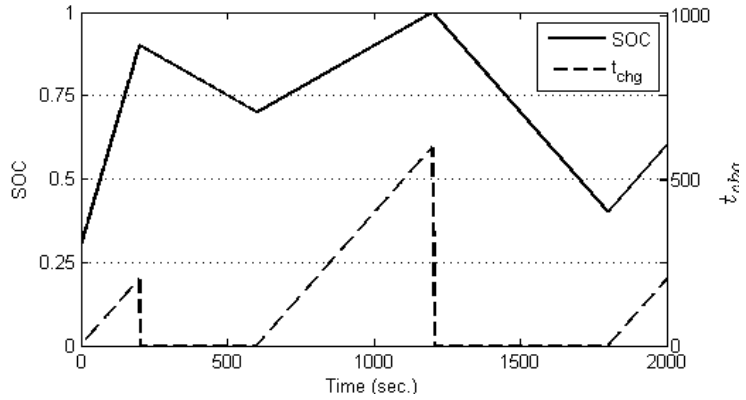
Having the complementarity constraints given by (2.7), one can model t_{chg} as the following differential inclusion:

$$f_{15} = \frac{\mathrm{d}t_{chg}}{\mathrm{d}t}(t) - \Big(\gamma_1 - 1000(1-\gamma_1)\Big)\gamma_2. \tag{2.8}$$

(2.7) and (2.8) show that when $t_{chg} \geq 0$, its value increases with a unity rate if the battery is in charging mode, i.e. $\gamma_1 = 1$. Once the battery switches to the discharging mode, i.e. $\gamma_1 = 0$, the value of t_{chg} falls down at a rate of -1000 until t_{chg} reaches zero when it remains constant afterward. Given (2.8), one can use (2.9) to model cycle life, $cycle_L$, as a continuous differential state. It is based on the fact that cycle life equals to $\sum_{n=1}^{\infty} \frac{1}{EFC(DOD_n)}$, where n is the number of discharging to charging mode



(a)



(b)

Figure 2.2: (a) Discontinuous cycle life and its equivalent continuous approximation; and (b) the introduced charging cycle, t_{chg}, state of battery in terms of state of charge (SOC).

switches and EFC depends on the value of $DOD_n = 1 - SOC_n$ at which battery switches to its charging mode. Fig. 2.2a illustrates the resulting continuous cycle life comparing with the original discontinuous value. In Fig. 2.2a, it can be seen that the value of cycle life becomes steady a little while after switching to charging mode, i.e. $t_{chg} \approxeq 20$, or during discharging mode, i.e. $\gamma_1 = 0$.

$$f_{16} = \frac{\mathrm{d}cycle_L}{\mathrm{d}t}(t) - \gamma_1 exp(-t_{chg})\left\{ \overbrace{\left(1 - SOC\right) \underbrace{\frac{1159SOC^2 - 952SOC + 745.3}{-SOC + 1.05909}}_{\text{Number of cycles}}}^{\text{Equivalent full cycle}} \right\}^{-1}. \tag{2.9}$$

2.2.3 Wind branch

The wind turbine performance can be measured according to the power coefficient curve, C_p, [9] with respect to tip speed ratio, λ (2.10a), and pitch angle β. (2.10c)

gives C_p curve of a wind turbine [94] consisting of three blades with radius, Rad, of 4.01 meter. Such a wind turbine generates the maximum power coefficient of 0.48 at the optimum tip speed ratio of 7.35.

$$f_{17} = \lambda - \frac{Rad \times \omega_r}{U_x}, \tag{2.10a}$$

$$f_{18} = \lambda_i - (\frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1})^{-1}, \tag{2.10b}$$

$$f_{19} = C_{p,norm} - \frac{1}{C_{p,max}}(C_1(\frac{C_2}{\lambda_i} - C_3\beta - C_4)exp(-\frac{C_5}{\lambda_i}) + C_6\lambda). \tag{2.10c}$$

The experimental coefficients $C_1 - C_6$ are defined in Table 6.1. λ_i in (2.10b) is an intermediate variable to make the equations easier to understand.

All fast voltage and current dynamics, except the angular velocity of the generator given by (2.11a), can be ignored for the long-term simulation and outer controller applications. For the sake of simplicity, it is assumed that there is no mechanical and electrical losses in the powertrain. Therefore, the electromagnetic power given by (2.11b) is equal to the output electrical power of wind branch.

$$f_{20} = \frac{\mathrm{d}\omega_r}{\mathrm{d}t}(t) - \frac{1}{J}(T_e - T_m - F\omega_r), \tag{2.11a}$$

$$f_{21} = -T_e \times \omega_r - I_{wtdc} \times V_{dc}, \tag{2.11b}$$

$$f_{22} = -T_m \times \omega_r - C_{p,norm}(\frac{U_x}{U_{x,base}})^3 P_{nom}, \tag{2.11c}$$

$$V_{LL} = \frac{1}{\sqrt{2}} k_E \omega_r, \tag{2.11d}$$

$$k_E = \sqrt{3} P \psi. \tag{2.11e}$$

The shaft inertia J $(Kg.m^2)$ and combined viscous friction coefficient F $(N.m.s)$ of PMSG are given by manufacturers. The V_{LL}, i.e. r.m.s. value of the line-to-line output voltage of generator given by (2.11d), depends on the mechanical angular velocity. Having number of pole pairs P and flux linkage ψ $(V.s)$ (see Table 6.1), one can use (2.11d) and (2.11e) to calculate the r.m.s. value of the line-to-line output voltage, V_{LL}, and the voltage constant k_E of generator.

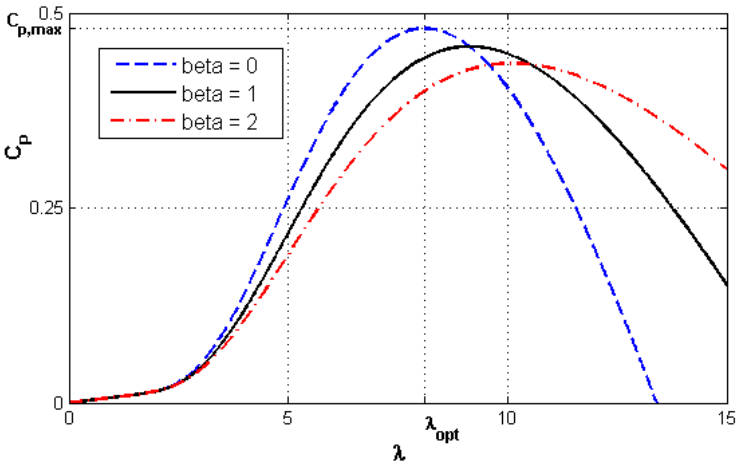


Figure 2.3: Power coefficient curves in terms of tip speed ratio.

Similar to battery and solar branches, buck converters can be modeled in steady-state CCM:

$$f_{23} = V_{dc} - D_w V_{wt}, \tag{2.12a}$$

$$f_{24} = I_{wt} - D_w I_{wtdc}. \tag{2.12b}$$

where D_w is switching duty cycle of converter and all remaining parameters are as in Figure 1.1.

The average dc output voltage of rectifier, V_{wt}, in the presence of the non-instantaneous current commutation is calculated as follows [13]:

$$V_{wt} = 1.35 V_{LL} - \frac{3}{\pi} P \omega_r L_s I_{wt}, \tag{2.13}$$

Replacing (2.11d), (2.11e), and (2.12) into (2.13) and rearranging, one has the following equation to calculate the dc output current of wind branch, I_{wtdc}:

$$f_{25} = I_{wtdc} - \frac{\pi}{3 P \omega_r L_s D_w} \left\{ \frac{1.35\sqrt{3} P \psi \omega_r}{\sqrt{2}} - \frac{V_{dc}}{D_w} \right\}. \tag{2.14}$$

2.3 Modelica model of stand-alone dc microgrids

In order to simulate the stand-alone dc microgrid shown in Figure 1.1, the presented mathematical model in Section 2.2 is declared with Modelica modeling language

Table 2.1: Outline of the developed Modelica model. MCPs and DIs are respectively indicated by † and ‡.

class StandaloneMicrogrids

...;

discrete Boolean gamma1 "true: charging";

Real soc(start = 0.6, fixed = true) "State of charge of battery";

Real tchg "The period of time that battery is in charging state";

RealInput Ux "wind speed (m/s)";

RealInput beta "pitch angle (degree)";

RealInput Dw "buck converter duty-cycle [0,1]";

equation

$Ipv = Iph - I0 * (exp((Ipv * rs\_Tc + Vpv)/a\_Tc) - 1) - (Ipv * rs\_Tc + Vpv)/rsh\_Tc\_Sx$;

$Iph = (Npvp * ((Rsh + Rs)/Rsh * Isc\_stc + Ki * (Tc - Tc\_stc)) * Sx)/Sx\_stc$;

$I0 = (Npvp*(Isc\_stc+Ki*(Tc-Tc\_stc)))/(exp(((Voc\_stc+Kv*(Tc-Tc\_stc))*q)/(nD*Ns*K*Tc))-1)$;

$Vpv = Vdc * (1 - Ds)$;

$Ipvdc = (1 - Ds) * iPV$;

$gamma1 =$ **if** $Ibstack \leq 0$ **then** true **else** false; (†)

$der(If) = -1/Ts * If + Ibstack/(Ts * Nbatp)$;

$der(Qact) = 1/3600 * Ibat$;

$der(Vexp) =$ **if** $gamma1$ **then** $P2/3600 * abs(Ibstack/Nbatp) * (P3 - Vexp)$ **else** $-(P2 * abs(Ibstack/Nbatp))/3600 * Vexp$; (‡)

$soc = 1 - Qact/Cmax$;

$Vbstack/Nbats =$ **if** $gamma1$ **then** (‡)

$V0 - R * Ibstack/Nbatp - (P1 * Cmax)/(Cmax - Qact) * Qact - (P1 * Cmax)/(Qact + 0.1 * Cmax) * If + Vexp$

else $V0 - R * Ibstack/Nbatp - (P1 * Cmax)/(Cmax - Qact) * Qact - (P1 * Cmax)/(Cmax - Qact) * If + Vexp$;

$der(tchg) =$ **if** $gamma1$ **then** 1 **elseif** $tchg > 0$ **then** -1000 **else** 0; (†‡)

$der(cycleL) =$ **if** $gamma1$ **then** $100/((1 - soc) * (1159 * soc\hat{}2 - 952 * soc + 745.3)/(-soc + 1.05909)) * exp(-tchg)$

else 0; (‡);

$-Te * \omega r = Iwtdc * Vbstack$;

$Ibstack = Ibatdc * (1 - Db)$;

$Vbstack = Vdc/(1 - Db)$;

$-Tm * \omega r = Cp\_pu * (Ux/12)\hat{}3 * Pnom$;

$der(\omega r) = (Te - Tm - F * \omega r)/J$;

$Iwtdc = pi * (1.35 * P * psi * sqrt(3) * \omega r)/sqrt(2)/(3 * Lst * P * \omega r * Dw)$;

$lambda = (R * \omega r)/Ux$;

$lambda\_i = 1/(1/(lambda + 0.08 * beta) - 0.035/(beta\hat{}3 + 1))$;

$Cp = (C1 * (C2/lambda\_i - C3 * beta - C4) * exp(-C5/lambda\_i) + C6 * lambda)/0.48$;

$Vdc = Iload * RL$;

$Ipvdc + Ibatdc + Iwtdc = Iload$;

end StandaloneMicrogrids;

[23]. The proposed acausal model, as HDAE equations, covers both continuous- (either differential or algebraic) and discrete-time behaviors [22]. Table 2.1 summarizes the developed Modelica model.

In Table 2.1, the MCPs given in (2.5) and (2.7) are indicated by symbol †. They are translated into IF-THEN-ELSE Modelica propositions. Moreover, DIs in (2.6a), (2.6d), (2.8), and (2.9) marked with symbol ‡, are also translated into IF-THEN-ELSE propositions. In order to minimize the number of equations in this study and unlike usual object-oriented approach, a flat HDAE model of stand-alone microgrids is proposed. It is important to note that the model is of Filippov type and therefore there is no need to re-initialize any state variable after mode switching. Otherwise, such re-initializations must be implemented using $change(.)$, $pre(.)$, and $reinit(.)$ facilities of Modelica.

2.4 Summary

In this chapter, stand-alone dc microgrids, consisting of wind, solar, and battery branches, has been mathematically modeled as complementarity dynamical systems. The developed mathematical model is in the form of hybrid DAEs of Filippov type, including mixed complementarity problems in order to model multi-modal operation as well as discontinuous cycle life state of the battery bank.

The proposed model has also been declared as Modelica models which is employed in Chapter 5 for long-term simulation of stand-alone dc microgrid operations. Moreover, the mathematically presented dc microgrid is embedded in NMPC strategies (see Chapter 6) as the system model to predict its behaviours.

Chapter 3

Stochastic Model Identification of Wind Speed and Solar Irradiance

Stochastic modelling of the uncertainties in renewable energy resources is applicable to design a reliable site and a robust controller for standalone sustainable microgrids. This chapter proposes stochastic models of hourly wind speed and solar irradiance for four locations across the UK. The goodness-of-fit of the proposed model is evaluated using Kolmogorov-Smirnov (K-S) test. The proposed model can be employed to simulate the amount of hourly wind speed and insolation for these locations.

3.1 Introduction

Wind kinetic energy and solar irradiance are assumed as energy resources which have considerable share in near future electricity production [47]. A typical standalone and sustainable microgrid employs an array of Photovoltaic (PV) modules and a Wind Turbine (WT) to supply load demands. There is also a battery bank to reduce power fluctuations. Despite employing proper sized components there still may be blackout periods because of uncertainties in energy resources which need to be considered through appropriate models.

The work in [40] has introduced a stochastic model for hourly global insolation absorbing by a PV array based on beta distribution. Although, it is a considerable achievement in stochastic modeling of insolation, the goodness-of-fit of the model

depends on location and the month of year [41], [95]. In particular, global solar irradiance is not necessarily fitted well for some months depending on location. The work in [40] has also introduced a stochastic model for hourly wind speed based on Weibull distribution. The fitness of hourly wind speed to Weibull distribution slightly depends on location and the month of year; nevertheless, it indicates an appropriate model for wind speed [96].

In this chapter, a stochastic model for hourly solar irradiance of four locations in the UK has been extracted and evaluated. The Maximum Likelihood Estimator (MLE) method has been employed to estimate the model parameters based on meteorological records. The goodness-of-fit of the estimated parameters has been evaluated using K-S test.

3.2 Problem formulation

3.2.1 Stochastic model of hourly solar irrandiance

Extraterrestrial solar irradiance can be estimated for any particular location with regard to its altitude and longitude as well as azimuth angle. However, the absorbed energy on an inclined surface is not equal to the terrestrial radiation but is a fraction of it. The difference between the terrestrial and absorbed insolation is mainly because of clouds which is a random phenomenon [97]. With attention to the probabilistic behavior of insolation, the amount of the electrical power generated with a PV array is a stochastic process, i.e., the value of hourly generated power is a random variable with a distribution related to the distribution of absorbed insolation.

The authors in [40] used a stochastic model for global insolation absorbed by a PV array. They modeled hourly global solar irradiance as beta probability distribution function (PDF) as follows:

$$f_{I_x}(i_x;\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} \left(\frac{i_x}{I_0}\right)^{\alpha-1} \left(1-\frac{i_x}{I_0}\right)^{\beta-1}. \tag{3.1}$$

where:

i_x	The solar irradiance (W/m^2)
$\Gamma(.)$	Gamma function
α and β	The shape parameters of beta distribution
I_0	The extraterrestrial solar irradiance (W/m^2)
$f_{I_x}(i_x)$	PDF of solar irradiance random variable I_x

The above equation introduces an hourly random variable which is the ratio of global to extraterrestrial solar irradiance of that hour. This random variable is defined as the clearness index. The authors in [98] proposed (3.2) to calculate extraterrestrial radiation for a particular location on earth.

$$I_0 = 1367 \left(\frac{\overline{D}}{D}\right)^2. \tag{3.2}$$

where the yearly average of the extraterrestrial radiation I_0 is $1367(w/m^2)$. The fraction $\frac{\overline{D}}{D}$ indicates the deviation of the daily distance D between sun and a particular location on earth from the yearly average distance $\overline{D}$ between sun and earth. This deviation varies through year by $\pm 3\%$ as earth orbits sun and can be approximated by a sinusoidal series as follows [98]:

$$\left(\frac{\overline{D}}{D}\right)^2 = 0.00011 + 0.034221\cos(\beta) + 0.00128\sin(2\beta) + 0.000719\cos(2\beta) + 0.000077\sin(2\beta), \tag{3.3a}$$

$$\beta = \frac{2\pi n}{365}. \tag{3.3b}$$

where β is the rotational angle of earth around sun and n represents the day of year in range of $[1, 365]$ or $[1, 366]$ for leap year.

Figure 3.1 illustrates the variation of extraterrestrial irradiance through year. In this figure, it can be seen that while earth is in the closest distance to sun during December and January, it is in the farthest distance during July.

Figure 3.2 illustrates different curves of beta distribution for different values of the shape parameters α and β. From Figure 3.2, it can be seen that by having the values of the shape parameters for a particular location, one can estimate the solar irradiance distribution for that location as follows:

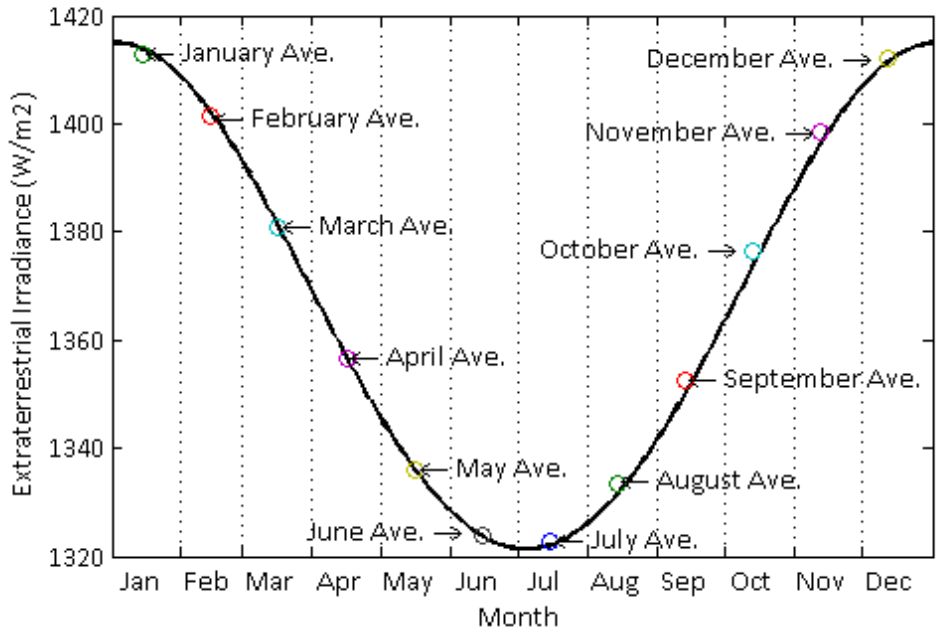


Figure 3.1: Extraterrestrial irradiance variations over a year with its monthly average values.

- $\alpha > 1 \ \& \ \beta < 1$: In this special case, hourly solar irradiation is mostly around extraterrestrial insolation of that day which is physically impossible.
- $\alpha < 1 \ \& \ \beta > 1$: It means that hourly solar irradiation is mostly around zero.
- $\alpha < 1 \ \& \ \beta < 1$: This particular case illustrates that hourly solar radiation is distributed almost uniformly.
- $\alpha > \beta > 1$: It models the case in which the amount of hourly solar irradiation is mostly greater than half of the extraterrestrial insolation of day.
- $\beta > \alpha > 1$: In contrast to the case of $\alpha > \beta$, this case models situations in which the amount of hourly solar irradiation is mostly less than half of the extraterrestrial insolation of day.

The shape parameters α and β of hourly global solar irradiance can be modeled when there are adequate meteorological records for that hour.

3.2.2 Stochastic model of hourly wind speed

The kinetic energy absorbed by a wind turbine depends on speed and direction of wind which are both random variables. Therefore, similar to PV arrays, the hourly generated power by wind turbines is a stochastic process.

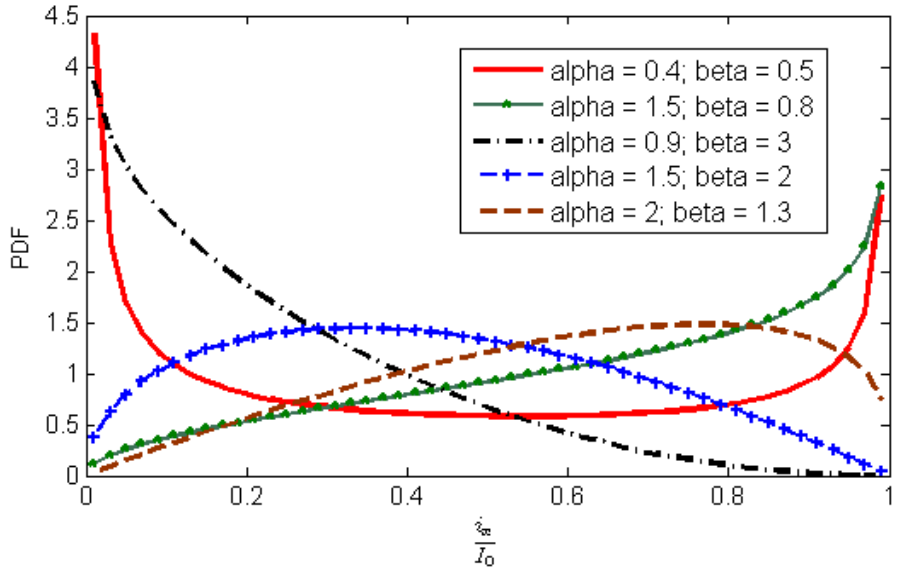


Figure 3.2: Different curves of beta distribution subject to different values of α and β.

The prior researchers in [40, 97, 96] presented stochastic models for wind speed. The authors in [96] employed Weibull probability distribution function (PDF) to model hourly wind speed as follows:

$$f_V(v;k,c) = \frac{k}{c}\left(\frac{v}{c}\right)^{k-1} \exp\left\{-\left(\frac{v}{c}\right)^{k}\right\}. \tag{3.4}$$

where:

v	Hourly wind speed (m/s)
k	The shape parameter
c	The scale parameter
$f_V(v)$	PDF of wind speed random variable V

Figure 3.3 shows different curves of Weibull distribution for different values of shape and scale parameters. Based on these curves, the wind speed distribution can be roughly estimated as follows:

- $c \leq 1$: In this case, hourly wind speed is mostly around zero. The greater value of $k > 1$, the narrower PDF around zero would be obtained.
- $c > 1$ & $k > 1$: In this special case, the distribution of hourly wind speed is a bell shape around an average value. The bell moves forward by increasing k or c.
- $k = c \gg 1$: It illustrates that hourly wind speed is mostly around the maximum value of that hour.

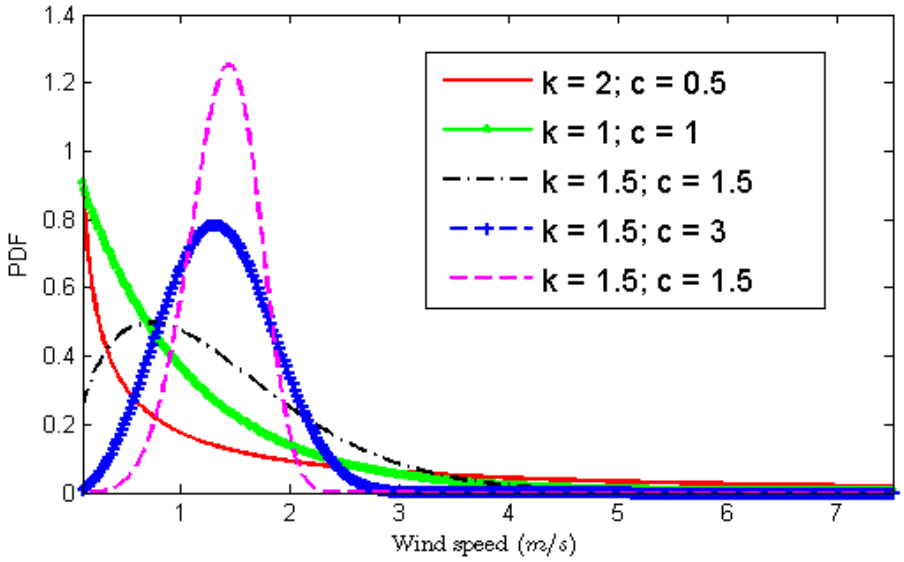


Figure 3.3: Different curves of Weibull distribution subject to different values of k and c.

The shape and scale parameters k and c of hourly wind speed can be modeled when there are adequate meteorological records for that hour.

3.3 Models assessment for UK locations

The empirical global solar irradiance and wind speed data samples for the UK are provided by British Atmospheric Data Center (BADC) [99]. In this study, hourly global solar irradiance and wind speed recorded by BADC for four locations across the UK, respectively over the period 1990 to 2010 and 2000 to 2008, is used. Table 3.1 illustrates these four locations and their geographical locations. It includes sites in Scotland, Wales, England and Shetland as one of the closest places in the UK to the North Pole.

MLE method is employed to estimate the probability distribution function and its parameters. The basic idea of MLE method as an optimizing algorithm is to maximize

Table 3.1: Locations across the UK studied in this thesis.

Location Name	Geographical Location
Shetland	Far North of Scotland
Argyll	South of Scotland
Dyfed	West of Walse
Cornwall	Southwest of England

the likelihood function with respect to parameters α and β or k and c, respectively, for hourly solar irradiance or wind speed.

3.3.1 Solar irradiance stochastic model parameter estimation

There are only 21 solar irradiance measurements for each hour over the period 1990 to 2010 which are not enough to calculate shape parameters. To increase the number of samples it is assumed that all observations for the same hour throughout any month are independent and identically distributed (IID). This assumption increases the number of data samples for each hour to around 600 samples. For instance, it is assumed that shape parameters of solar irradiance distribution during the period 10:00 to 11:00 hour of all days in January are the same. However, it does not mean that the amounts of solar irradiance of all days in January for this period of time are similar. Furthermore, monthly average values of extraterrestrial solar irradiance (Figure 3.1) have been applied to all hours of that month.

Figure 3.4 illustrates graphically the calculated shape parameters of hourly solar irradiance for the four locations summarized in Table 3.1. The exponents for each location have been extracted employing MLE method with around 600 hourly records coupled with monthly average extraterrestrial irradiance. The hourly solar irradiance can be roughly estimated using values of α and β from Figure 3.4 and comparing them with different curves in Figure 3.2. For instance, Figure 3.4 (g) and (h) show that values of α and β of Argyll in south of Scotland are almost always less than unity. From Figure 3.2, it can be seen that such values for α and β means that solar irradiance has a uniform distribution. In other words, solar irradiance can be any value with the same probability.

3.3.2 Solar irradiance stochastic model assessment

K-S statistical test has been employed in this study to evaluate goodness-of-fit of the proposed model. According to this method, goodness-of-fit is quantized with the maximum distance between empirical and estimated cumulative distribution functions (CDF).

Figure 3.5 depicts a comparative analysis between theoretical and empirical CDFs. For instance, for the time slot between 12:00 and 13:00 during October in Dyfed,

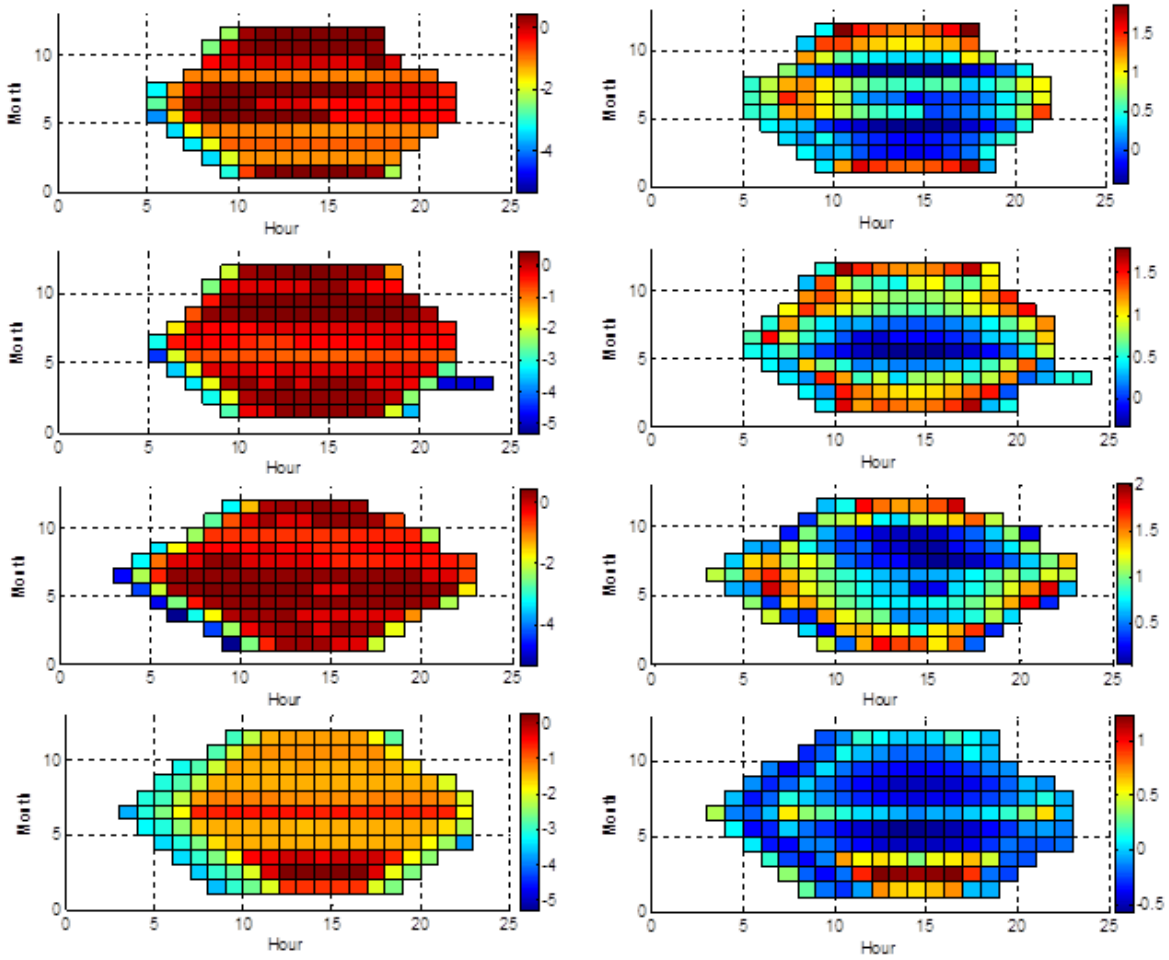


Figure 3.4: Estimated logarithmic values of (a) α and (b) β for Dyfed (west of Wales), (c) α and (d) β for Cornwall (southwest of England), (e) α and (f) β for Shetland (far north of Scotland), and (g) α and (h) β for Argyll (south of Scotland).

the maximum distance is 0.070 and is greater than the critical value of 0.053 for 651 samples. It means that the model of global solar irradiance, presented by [40], is not fitted well with meteorological records of Dyfed.

3.3.3 Wind speed stochastic model parameter estimation

Using the hourly wind speed measurements over the period between 2000 and 2008 and applying the same assumption under sub-section 3.3.1, one has 270 hourly wind speed samples instead of just 9 samples. Then, these 270 samples are used to calculate the hourly shape and scale parameters. Figure 3.6 illustrates graphically the calculated shape and scale parameters of hourly wind speed for the four locations indicated in Table 3.1.

The hourly wind speed can be roughly estimated using values of k and c extracted

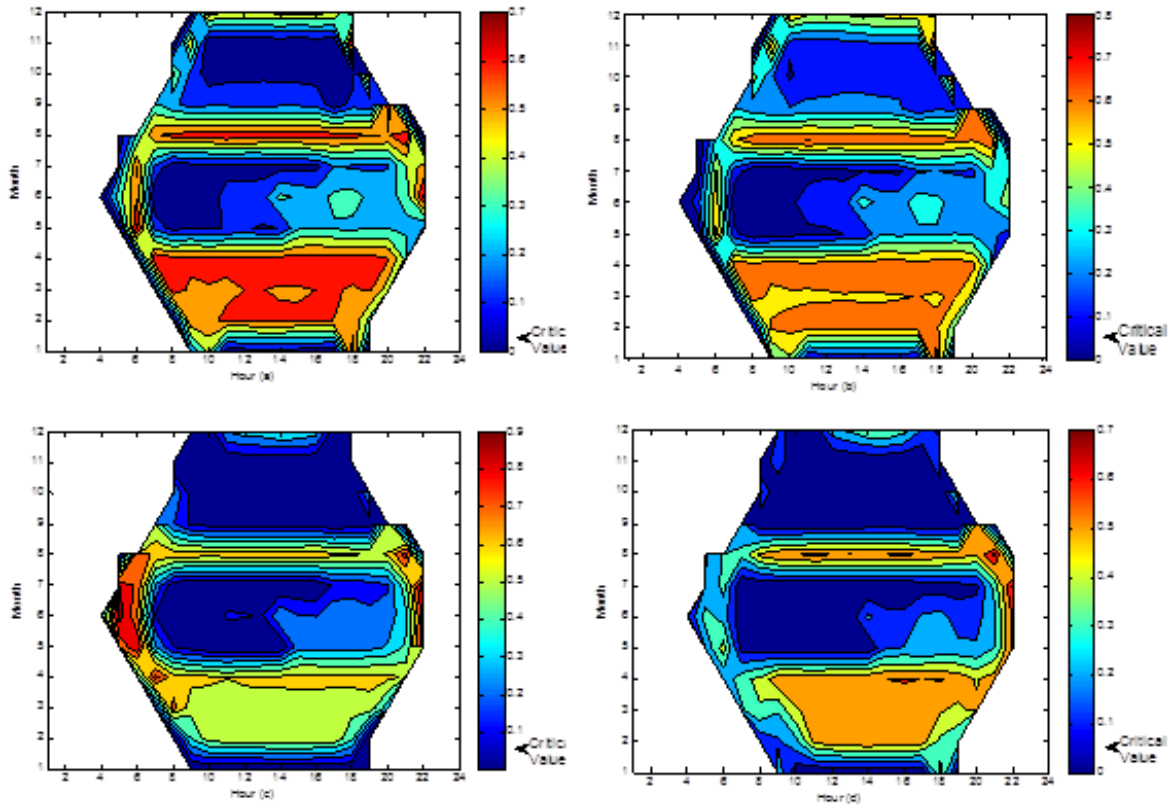


Figure 3.5: The maximum distance of theoretical and empirical CDFs and critical values for (a) Dyfed (west of Wales), (b) Cornwall (southwest of England), (c) Shetland (far north of England), and (d) Argyll (south of Scotland).

from Figure 3.6 and comparing them with different curves in Figure 3.3. For instance, Figure 3.6 (a) and (b) show that the value of k of Dyfed is less than two while the value of the parameter *c* is greater than five. From Figure 3.3, it can be seen that these values of the parameters *k* and *c* indicates a distribution of wind speed which is a narrow bell shape around average value.

3.3.4 Wind speed stochastic model assessment

Figure 3.7 depicts a comparative analysis between theoretical and empirical CDFs. For example, for the period 12:00 to 13:00 during October in Dyfed, the maximum distance is less than the critical value for 279 samples. It means that the estimated model of wind speed for this hour is fitted well with meteorological records.

3.4 Summary

The amount of electrical power generated by a hybrid sustainable microgrid is a stochastic process with respect to hourly solar irradiance and wind speed as a ran-

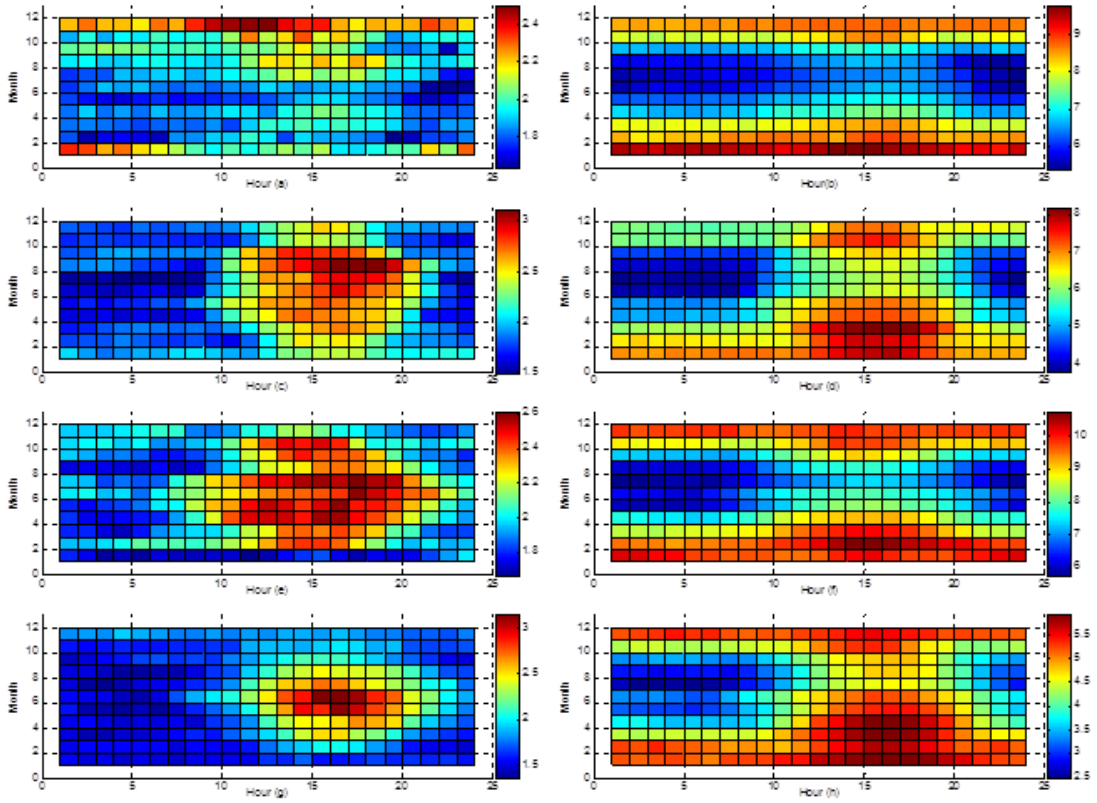


Figure 3.6: Estimated values of (a) k and (b) c for Dyfed (west of Wales), (c) k and (d) c for Cornwall (southwest of England), (e) k and (f) c for Shetland (far north of Scotland), and (g) k and (h) c for Argyll (south of Scotland).

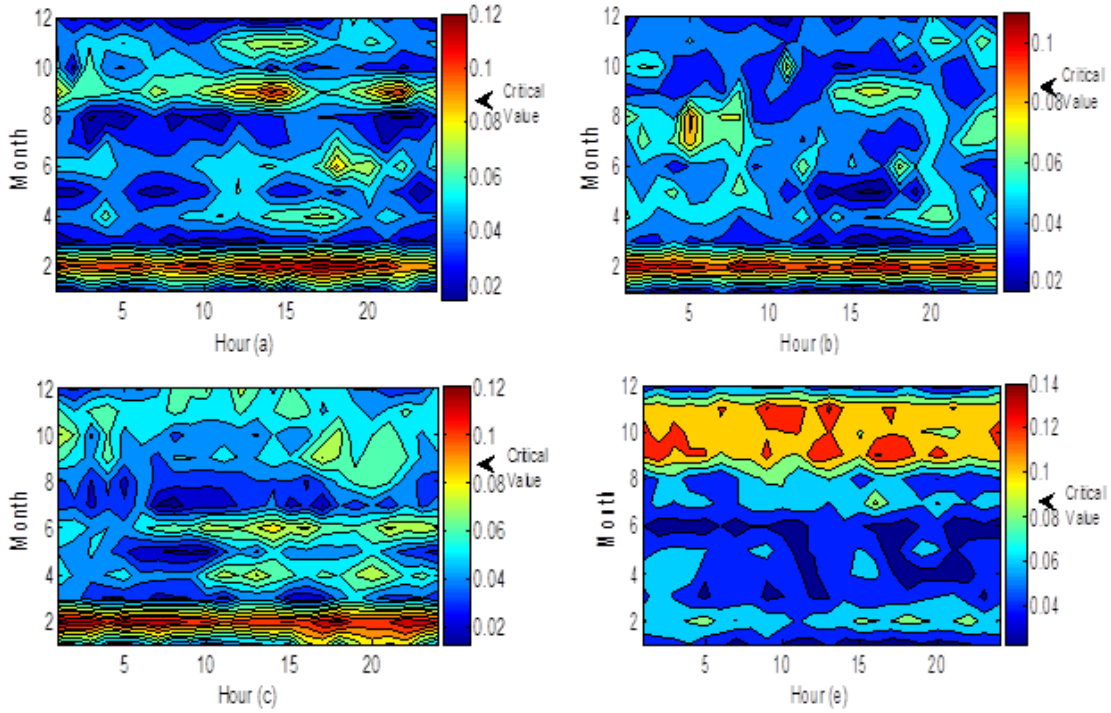


Figure 3.7: The maximum distance of theoretical and empirical wind speed CDFs and critical values for (a) Dyfed (west of Wales), (b) Cornwall (southwest of England), (c) Shetland (far north of England), and (d) Argyll (south of Scotland).

dom variables. In order to study the behavior of such microgrids or to design a reliable site for a location, it essentially requires modeling the probabilistic behavior of solar irradiance and wind speed.

This chapter has proposed a stochastic model of hourly solar irradiance and wind speed for four locations across the UK. The proposed model has been evaluated employing K-S test.

The proposed stochastic models are used in Chapter 5 to simulate the hourly wind speed and solar irradiance. The simulation results obtained for hourly wind speed indicate proper match with statistical moments of available measurements. Although the simulation results for hourly solar irradiance are not completely match with measurements, there are still two advantages to gain by employing stochastic model, which are simplicity of simulation and considering uncertainties due to cloud shading.

Chapter 4

Model Identification of Photovoltaic (PV) Modules

Due to simulation requirements as well as recent applications of model-based controllers, accurate and fast PV model identification and maximum power point (MPP) estimation methods are becoming essential to reduce PV power losses effectively. The classical PV model identification methods use empirical data of MPP at the standard test condition (STC). However, the nominal operating cell temperature (NOCT) is the more practical condition and it is shown that the extracted model is not well suited to it. The proposed method in this chapter estimates an accurate equivalent electrical circuit for PV modules using both STC and NOCT information provided by manufacturers. A multi-objective global optimization problem is formulated using only the main equation of PV module at these two conditions that restrains the errors due to employing experimental temperature coefficients. A novel combination of a genetic algorithm (GA) and the interior-point method (IPM) allows the proposed method to be fast and accurate regardless PV technologies. It is shown that the overall identification error, which is defined by the sum of MPP errors of both STC and NOCT conditions, is improved by a factor between 5.1% to 31% depending on PV technologies.

4.1 Introduction

Advances in PV technologies during the last decade have increased the share of solar energy in the growing electricity market. PV modules are non-linear and complex

still very popular components [100], [55] since they are easy to install and operate. However, it is essential to model PV module accurately to be able to predict its behavior at different operating conditions. While the PV behavioral prediction is essential for system design and PV module simulation, it can also be useful for a probable employing of model-based prediction controllers (MPC) to manage energy generation or to track MPP of operation. There are many studies conducted on different MPPT methods (e.g. see [101–104]).

A *PV cell* is a P-N junction that is typically modeled with an equivalent electrical circuit [47]. Although there are prior researches focused on non-electrical models based on either artificial neural network (ANN) [105] or adaptive neuro-fuzzy inference scheme (ANFIS) [106], equivalent electrical circuit is the most popular PV model. The core advantage of the electrical models is that unlike the other models it does not necessarily require measured data samples to be trained. Moreover, it can easily be simulated with other components as an integrated electrical circuit. Figure 4.1 illustrates a single-diode equivalent electrical circuit of PV cell, which is also applicable to model a *PV module*. A PV module consists of several PV cells connected together in series. Although double-diode circuit provides better modeling of the loss in depletion region caused by recombination of carriers [107, 108], single-diode model still offers a good compromise between accuracy and simplicity [46]. A *PV array,* which is a combination of several photovoltaic modules in series and parallel arrangement, can be modeled with the same circuit, too. There are six parameters in this model, namely, stray capacitor C, ideality factor of diode n_d, photocurrent I_{ph}, reverse saturation current of diode I_0, series resistor R_s, and shunt resistor R_{sh}. While stray capacitor C is neglected [48], there are still five unknown parameters for each PV module, which require being determined with an identification method.

To the best of the authors knowledge, the prior researches on PV electrical equivalent model identification methods can be categorized into three groups. Each of these methods can also be applied to either a simplified model taken the shunt resistor as infinite [109, 110] or the normal circuit leading to a four- or five-parameter estimation problem, respectively.

In the first group, the authors in [111–115] presented analytical methods to calculate parameters. The identification method proposed in [111] attempts to approximate

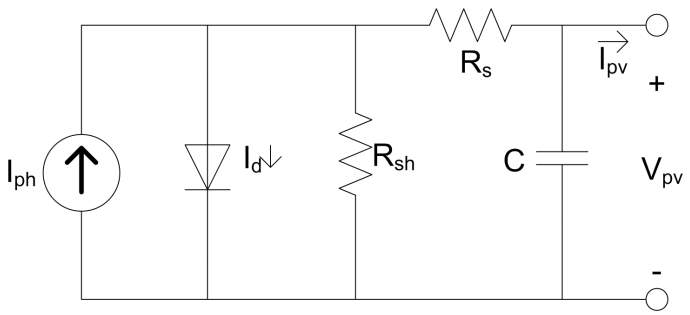


Figure 4.1: Single-diode equivalent electrical circuit of PV cells, modules, and arrays.

and solve analytically the equation of PV as a polynomial of the first order. Even though a simple parameter estimation method is demonstrated, the method does not provide enough accuracy. The works of [112–115], however, introduced a system of five equations to resolve five model parameters. Three of these equations are extracted from three remarkable points [46], i.e., short- and open-circuit points coupled with MPP. While the forth equation is constructed based on the fact that derivative of power at MPP is zero, the fifth equation is more challenging. The authors in [112] employed an approximation for temperature coefficient as the fifth equation, however, the others adopts the equation introduced in [113] as a simplified approximation of shunt resistor. Although these analytical methods are straightforward and only need nominal values from datasheets, they still need sophisticated numerical or even optimization methods to solve the resulting non-linear and transcendental system of equations. Moreover, the accuracy is deteriorated as a result of using simplified approximation of either temperature coefficient or shunt resistor.

Alternatively, there are prior researches focused on identifying a PV current-voltage ($I - V$) curve (Figure 4.2) being the best fit to empirical data [116–119]. The authors in [116] and [117] proposed a genetic algorithm as a curve fitting procedure to identify the circuit in Figure 4.1. The main issue of this approach is that it relies on measured data rather than information available in datasheets. Moreover, it requires measuring cell temperature, which is a cumbersome task, of PV module. With the aim of avoiding the need to measure points on different $I - V$ curves, the authors in [118] introduced a set of equations to calculate remarkable as well as two intermediate points at different values of insolation and cell temperature. This method is adopted by Sandia national laboratories (SNL), USA [120] as the reference method in PV model identification. The equations rely on several empirical parameters, which need to be extracted for each type of PV module. The national

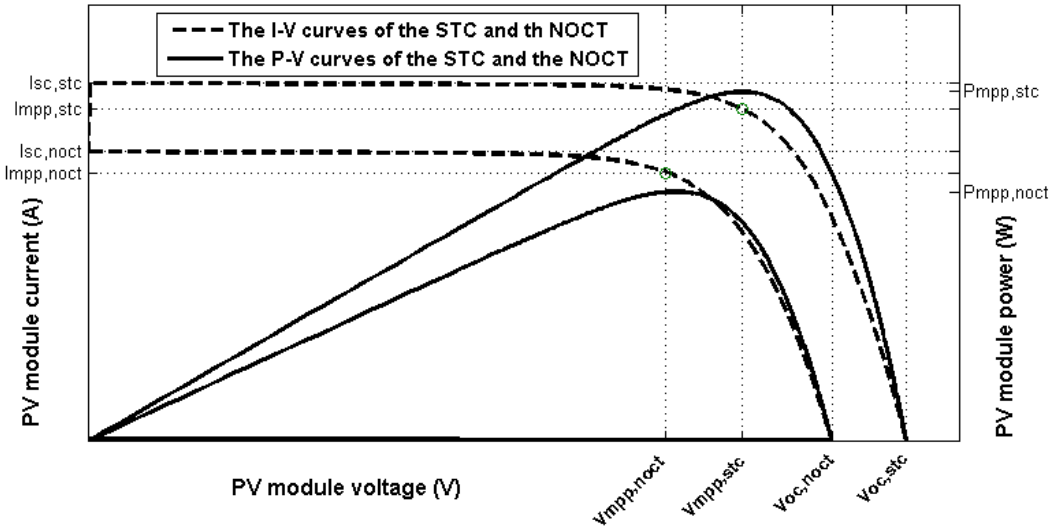


Figure 4.2: $I - V$ curves of PV module for STC and NOCT conditions.

institute of standards and technology (NIST), USA [121] provides these empirical parameters for different PV modules technologies [112]. The procedure needs to be repeated for any new PV module type.

Finally, in order to avoid issues of these approaches, i.e., the needs for measuring points on $I - V$ curves and degrading the accuracy caused by simplifications, the authors in [46], and [92] proposed methods based on optimization techniques. The identification method proposed in [46] employed an exhaustive search algorithm to explore all values of ideality factor and series resistor. It calculates the value of shunt resistor iteratively. Although it is an effective algorithm, it requires checking all possible values of ideality factor and series resistor before determining the optimum ones as well as calculating a complete $I - V$ curve at each iteration. These two issues make the procedure quite slow. Furthermore, the speed of procedure is sensitive to the resolution of parameters with an order of $O\left(\left(\frac{resolution_{new}}{resolution_{old}}\right)^2\right)$. The authors in [92] reformulated the problem as an unconstrained optimization problem. Then, they employed the particle swarm optimization (PSO) technique to solve it at different values of cell temperature. Although the formulation and employed algorithm improve the execution speed of the procedure, it still requires solving $I - V$ as well as two other transcendental equations at each iteration.

The main limitations of these methods are that they either require sophisticated experimental setups or use only STC information available in datasheets. It is difficult to measure experimental points with regard to the difficulty in measuring cell tem-

perature. Meanwhile, STC information estimates parameters with good accuracy for MPP at STC; however, it is shown in Section 4.3.2 that it introduces a considerable error for MPP at NOCT. Furthermore, the methods based on the optimization techniques solve repeatedly the non-linear transcendental $I-V$ curve for the whole range of the PV voltage and it causes them to be slow. To overcome these limitations, this chapter proposes a novel PV model identification method to estimate the electrical parameters with minimum *overall error*, namely, *overall optimum solution*. The overall error is defined by the sum of the STC and NOCT errors, which are the MPP estimation errors for the STC and NOCT curves, respectively. In this chapter, the proposed method is applied for extracting the electrical parameters of the equivalent single-diode circuit; however, with some modifications it is also applicable to the double-diode model of a PV module. The proposed method relies on a multi-objective optimization problem (MOOP) using either any empirical inaccurate equations or any assumptions on parameters. A novel combination of a GA and IPM is introduced to solve the resulting MOOP in considerably short time. Moreover, it is shown that applying the proposed STC-based method, which considers the STC error rather than overall error as objective function, the sum of the voltage, current, and power errors is significantly low.

The results are validated against experimental benchmark curves, available in PV module datasheets. They are also compared with the results of two prior researches [46], [92] with respect to both the overall and STC errors. Applying the proposed method for Kyocera KC200GT and SHELL ST40 PV modules and comparing with the results in [46] and [92] shows substantial improvements. Employing IPM instead of calculating complete $I-V$ curves, the proposed method is around 10 times faster than the method introduced in [46].

This chapter is organized as follows: the next section provides a summary of PV mathematical model. Section III presents a formulation of MPP estimation and parameter identification problems as well as the proposed procedure. The identified electrical parameters for three different PV technologies coupled with their validation are given in Section IV. Finally, Section V presents a conclusion of the study. Figure 4.3 provides a summary of the chapter indicating different stages of the PV model identification and validation procedure.

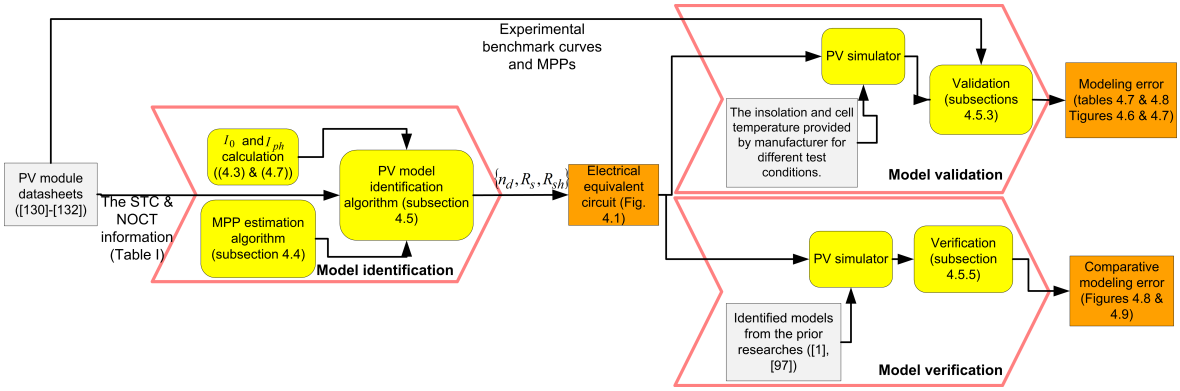


Figure 4.3: An outline of chapter indicating different stages of the proposed procedure.

4.2 PV equivalent electrical circuit

Figure 4.1 illustrates a single-diode equivalent electrical circuit of a PV module. Among electrical parameters, series resistor R_s is the sum of structural resistances [46] and shunt resistor R_{sh} representing the leakage current [46]. Applying Kirchhoff current law (KCL) to the junction point of these two resistors gives the characteristic equation of PV module, which is a non-linear transcendental equation, as follows:

$$I_{pv} = I_{ph} - I_0 \left\{ \exp\left(\frac{V_{pv} + R_s I_{pv}}{n_d N_s} \frac{q}{K T_c}\right) - 1 \right\} - \frac{V_{pv} + R_s I_{pv}}{R_{sh}}. \tag{4.1}$$

where V_{pv} and I_{pv} are, respectively, output voltage and current of PV module and all other symbols are defined as follows:

- q The electron charge (1.60218×10^{-19})
- K Boltzman constant (1.38066×10^{-23})
- N_s The number of PV cells in series constructing PV module (-)
- T_c The current amount of PV cell temperature (K)

Photocurrent I_{ph} and reverse saturation current of the diode I_0 for STC and NOCT conditions are calculated using available parameters in datasheets [46], [122]:

$$I_{ph} = \left(\frac{R_s + R_{sh}}{R_{sh}} I_{sc,X} + k_I \left(T_c - T_{c,X}\right) \right) \frac{S}{S_X}. \tag{4.2}$$

$$I_0 = \frac{I_{sc,X} + k_I \left(T_c - T_{c,X}\right)}{\exp\left(\frac{V_{oc,X} + k_V \left(T_c - T_{c,X}\right)}{n_d N_s} \frac{q}{K T_c}\right) - 1}. \tag{4.3}$$

where parameters are defined as follows:

$I_{sc,X}$	The short-circuit current of PV module at condition X (A)
k_I	The temperature coefficient of short-circuit current ($A/^\circ C$)
k_V	The temperature coefficient of open-circuit voltage ($V/^\circ C$)
S	The current amount of solar irradiance (W/m^2)
S_X	The standard amount of the solar irradiance at condition X (W/m^2)
$T_{c,X}$	The standard amount of the cell temperature at condition X (K)
$V_{oc,X}$	The open-circuit voltage of PV module at condition X (V)
X	The operating condition that is either STC or NOCT (-)

Manufacturers provide all above parameters, i.e., empirical values of the remarkable points for each operation conditions as well as cell temperature at NOCT. Table 4.1 illustrates these information for PV modules studied in this chapter.

Under STC and NOCT conditions, (4.2) and (4.3) can be rewritten as follows:

$$I_{ph} = \begin{cases} \dfrac{R_s + R_{sh}}{R_{sh}} I_{sc,stc} & \text{For STC-curve,} \\ \dfrac{R_s + R_{sh}}{R_{sh}} I_{sc,noct} & \text{For NOCT-curve.} \end{cases} \tag{4.4}$$

$$I_0 = \begin{cases} \dfrac{I_{sc,stc}}{\exp\left(\dfrac{V_{oc,stc}}{n_d N_s} \dfrac{q}{K T_{c,stc}}\right) - 1} & \text{For STC-curve,} \\ \dfrac{I_{sc,noct}}{\exp\left(\dfrac{V_{oc,noct}}{n_d N_s} \dfrac{q}{K T_{c,noct}}\right) - 1} & \text{For NOCT-curve.} \end{cases} \tag{4.5}$$

By substituting (4.4) and (4.5) into (4.1), one obtains the equations of the two $I - V$ curves in Figure 4.2, which their stationary points are MPPs.

There are other equations to model deviation of photocurrent, short-circuit current, and open-circuit voltage due to a variation in cell temperature [112], [92]. None of these equations are used in the proposed method and it only uses (4.1) under STC and NOCT conditions to increase the accuracy of the method.

4.3 MPP estimation and PV identification problems formulation

Basically, PV model identification can be interpreted as a parameter estimation problem (as a sub-class of optimization problems) with a few difficulties. It requires find-

ing a global optimum point in a problem with multiple local optima. Moreover, it is shown that existing two objectives, i.e. matching of MPP for each STC- and NOCT-curve against manufacturer data, are conflicting objectives. The PV model identification problem is formulated as a global MOOP with two constraints on MPPs of STC and NOCT curves. A proper GA is proposed to solve this global MOOP. It relies on a fast MPP estimation algorithm, explained below, that employs IPM.

4.3.1 MPP estimation problem formulation

MPP is a point (V_{mpp}, I_{mpp}) on $I-V$ curve that produces the maximum possible power out of PV module. As a result, the problem of estimating MPP for either STC- or NOCT-curve can be formulated as follows:

$$[V_{mpp}, I_{mpp}] = \text{arg} \underset{\mathrm{v_{pv}}, \mathrm{i_{pv}} \in \Re}{\text{minimize}} \left\{ \frac{1}{v_{pv} i_{pv}} \right\} \tag{4.6a}$$

subject to:

$$i_{pv} = i_{ph} - i_0 \left\{ \exp(\frac{v_{pv} + R_s i_{pv}}{n_d N_s} \frac{q}{K T_c}) - 1 \right\} - \frac{v_{pv} + R_s i_{pv}}{R_{sh}} \tag{4.6b}$$

$$i_{ph} = \frac{R_s + R_{sh}}{R_{sh}} I_{sc} \tag{4.6c}$$

$$i_0 = \frac{I_{sc}}{\exp(\frac{V_{oc}}{n_d N_s} \frac{q}{K T_c}) - 1} \tag{4.6d}$$

$$0 \leq [v_{pv}, i_{pv}]^T \leq [V_{oc}, I_{sc}]^T. \tag{4.6e}$$

Given PV module information for the test conditions in Table 4.1 and electrical parameters in Figure 4.1, MPP can be estimated for either of test conditions. The objective function, which needs to be minimized, is the reverse of generated power. There is a non-linear constraint (4.6b) that is the $I-V$ curve of PV module. Except box constraints in (4.6e), other constraints provide equations to calculate photocurrent and diode reverse saturation current.

4.3.2 PV model identification problem formulation

In this chapter, PV model identification problem has been extended as the procedure of identifying the best values for n_d, R_s, and R_{sh} parameters (Figure 4.1) such

that not only $I-V$ curve (in (4.1)) at STC condition but also appropriate $I-V$ curve at NOCT condition have the maximum output power as much similar as possible to $(V_{mpp}$,$I_{mpp})$ points reported by manufacturers (Table 4.1). Identifying these parameters, (4.2), and (4.3) can be employed to calculate the remaining I_{ph}, and I_0 parameters.

Table 4.1: PV modules studied in this thesis and their datasheet information.

Parameters	Polycrystalline KC200GT (Kyocera)	Thin film ST40 (SHELL)	Monocrystalline E20/333 (Sunpower)
$I_{mpp,stc}(A)$	7.61	2.41	6.09
$V_{mpp,stc}(V)$	26.3	16.6	54.7
$P_{mpp,stc}(W)$	200.143	40.00	333.123
$I_{sc,stc}(A)$	8.21	2.68	6.46
$V_{oc,stc}(V)$	32.9	23.3	65.3
$I_{mpp,noct}(A)$	6.13	1.88	4.91
$V_{mpp,noct}(V)$	23.2	14.7	50.4
$P_{mpp,noct}(W)$	142.216	27.7	247.64
$I_{sc,noct}(A)$	6.62	2.2	5.22
$V_{oc,noct}(V)$	29.9	20.7	61.2
$k_I(A/^\circ C)$	3.18×10^{-3}	0.35×10^{-3}	3.5×10^{-3}
$k_V(V/^\circ C)$	-1.23×10^{-1}	-100×10^{-3}	-176.6×10^{-3}
$N_s(-)$	54	36	96
$T_{c,noct}{}^\circ C$	47	47	45
$T_{c,stc}{}^\circ C$	25	25	25

Figure 4.4 illustrates *identification errors*, which are defined as the normalized Euclidean distance between the estimated MPP and experimental MPP provided by manufacturer, for different values of ideality factor. They have been depicted just up to the optimum points for simplicity. In Figure 4.4, it can be seen that for each individual value of ideality factor there is a local optimum with respect to other model parameters R_s and R_{sh} and this fact introduces multiple local optima to the problem. For example, when n_D equals to 1.25, which is illustrated by "$+$", local optima with the error of -8.056 locates at $(R_s, R_{sh}) = (0.2454\Omega, 408.4\Omega)$.

Applying identification error to each STC or NOCT information, respectively, intro-

Table 4.2: Conflicts between STC- and NOCT-based objectives.

PV modules $\longrightarrow$	KC200GT		ST40	
Error Type $\downarrow$	Parameters of [46]	Using NOCT-based objective	Parameters of [92]	Using NOCT-based objective
STC-identification error	0.00262	0.03192	0.00499	0.03214
NOCT-identification error	0.02401	0.00003	0.05166	0.00002

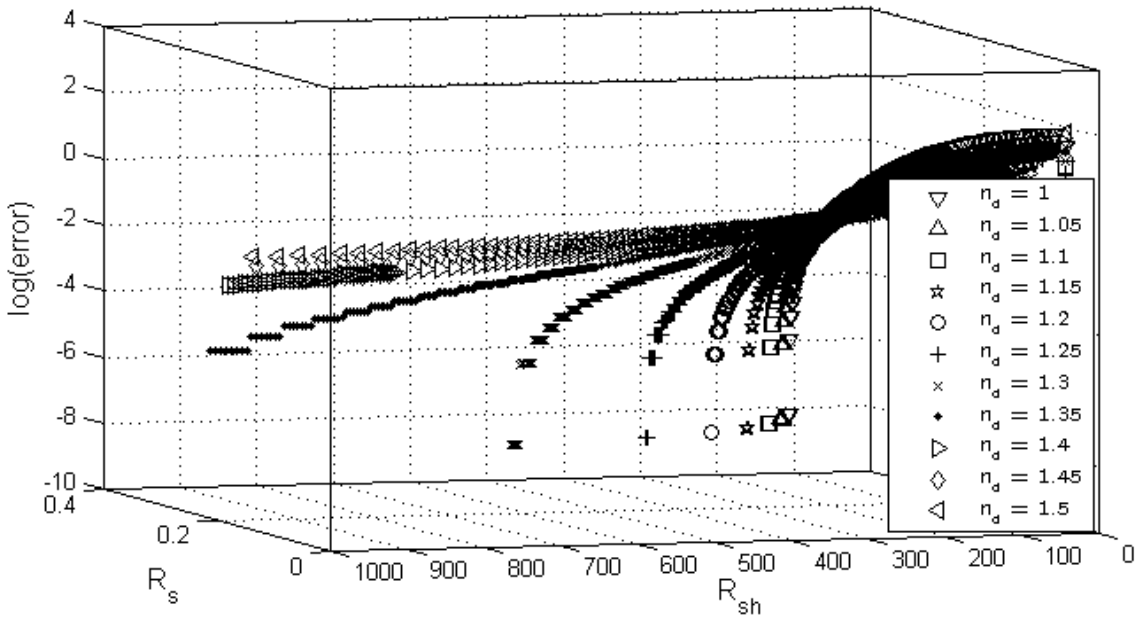


Figure 4.4: The existance of local optima for different values of ideality factor.

duces *STC-identification error* and *NOCT-identification error*. Table 4.2 illustrates corresponding errors to the prior estimated parameters in [46] and [92]. It also compares the amount of these identification errors with the case that parameters are identified with respect to MPP at NOCT condition. From Table 4.2, it can be seen that minimizing either of these errors does not minimize the other one. If the estimated parameters propose an optimum point for the STC-identification error, these parameters do not provide an optimum point for the NOCT-identification error and vice versa. Hence, considering both STC and NOCT conditions introduce a MOOP [123] with the following conflicting objectives:

$$J_1(v_{pv}, i_{pv}) = \|[\frac{v_{pv} - V_{mpp,stc}}{V_{mpp,stc}}, \frac{i_{pv} - I_{mpp,stc}}{I_{mpp,stc}}, \frac{v_{pv}i_{pv} - P_{mpp,stc}}{P_{mpp,stc}}]\|_2. \tag{4.7}$$

$$J_2(v_{pv}, i_{pv}) = \|[\frac{v_{pv} - V_{mpp,noct}}{V_{mpp,noct}}, \frac{i_{pv} - I_{mpp,noct}}{I_{mpp,noct}}, \frac{v_{pv}i_{pv} - P_{mpp,noct}}{P_{mpp,noct}}]\|_2. \tag{4.8}$$

$$J(v_{pv}, i_{pv}) = \{w_1\theta_1 J_1(v_{pv}, i_{pv}) + w_2\theta_2 J_2(v_{pv}, i_{pv})\}. \tag{4.9}$$

Due to the fact that power is the multiplication of voltage and current, considering only the generated power in objective functions and discarding these two factors introduces an algorithmic error as follows:

$$\varepsilon_p = V_{mpp}\varepsilon_i + I_{mpp}\varepsilon_v. \tag{4.10}$$

(4.10) indicates that even small amount of round-off error in the estimated V_{mpp} and I_{mpp}, namely, ε_v and ε_i are magnified by the values of voltage and current before taking into account as the estimated P_{mpp} error, ε_p. In order to eliminate these issues, the difference between all these three factors and nominal values available in datasheets have been considered in (4.7) and (4.8).

Although open- and short-circuit points can also be added to objective functions, it is unnecessary because they are considered in (4.4) and (4.5). Figure 4.6 of the next section shows that the generated $I - V$ curve using estimated parameters passes through all remarkable points accurately.

There are two approaches to solve the resulting multi-objective optimization problems: i) converting to a single-objective problem; and ii) determining an entire Pareto solution [124]. While the former approach returns a single solution, the latter provide a Pareto optimal set of solutions that are not dominated with each other. The decision-maker selects among Pareto optimal set according to trade-offs. The proposed algorithm in this chapter without loss of generality employs the first approach for two reasons. While it is easier to implement, the proposed algorithm also aims at estimating the best set of values for parameters rather than a set of optimal solutions.

The individual objectives (4.7) and (4.8) are converted to a single objective (4.9), employing a weighted convex combination of the normalized objective functions $\hat{J} = \left[\hat{J}_1, ..., \hat{J}_k\right]$ with weights $w = [w_1, ..., w_k]^T$ [124]. Convex combination means that $\sum_{i=1}^{k} w_i = 1$ and the normalized objective function is defined as follows:

$$\hat{J} = [\theta_1, ...\theta_k]^T [J_1, ..., J_k] \tag{4.11a}$$

$$\theta_i = \frac{1}{J_i(x^{[i]})} \tag{4.11b}$$

$$x^{[i]} = \arg \underset{x \in \chi}{\text{maximize}} \quad \{J_i(x)\}. \tag{4.11c}$$

where k is the number of individual objective functions. In order to keep the values of all objective functions in the range of $[0, 1]$, the normalization factors $\{\theta_i\}$ are selected as the reverse of their maximum values. The resulting single-objective PV model identification problem is proposed as follows:

$$[R_{s,opt}, R_{sh,opt}, n_{d,opt}] =$$

$$\text{arg} \underset{\mathrm{R_s},\mathrm{R_{sh}},\mathrm{n_d}\in\Re}{\text{minimize}} \{w_1\theta_1 J_1(v_{mpp,stc}, i_{mpp,stc}) + w_2\theta_2 J_2(v_{mpp,noct}, i_{mpp,noct})\} \tag{4.12a}$$

subject to:

$$[v_{mpp,stc}, i_{mpp,stc}] = \text{The solution of (4.6) for STC} \tag{4.12b}$$

$$[v_{mpp,noct}, i_{mpp,noct}] = \text{The solution of (4.6) for NOCT} \tag{4.12c}$$

$$[R_{s,min}, R_{sh,min}, n_{d,min}]^T \leq [R_s, R_{sh}, n_d]^T \leq [R_{s,max}, R_{sh,max}, n_{d,max}]^T. \tag{4.12d}$$

Through this study, normalization factors are calculated dynamically and weights are assumed to be equal (i.e. $[0.5, 0.5]$). The proposed method for this case can be named as *full method*. The equal weights imply that there is no preferences between either of STC or NOCT conditions, however, if there is such a preference, the convex weights can be unbalanced. Particularly, applying either of the values of $[1, 0]$ or $[0, 1]$ convert (4.12) to a single (only (4.7) or (4.8)) objective problem and the resulting methods are respectively called *STC- and NOCT-based method*. Two constraints in (4.12b) and (4.12c) present MPPs of STC and NOCT curves. Table 4.3 summarizes box constraints of electrical parameters for different PV modules studied in this thesis.

4.4 MPP estimation algorithm

The optimization problem given in (4.6) can be solved employing either the curve-inspecting technique, a local optimization method, or even heuristic algorithms such as a GA. The authors in [108], [46], and [92] employed the curve-inspecting method, which is an exhaustive search method, to find MPP of the $I - V$ curve. It solves the

Table 4.3: Box constraints of PV module electrical parameters.

Parameters	KC200GT	ST40	E20/333
$n_{d,\min}(-)$	1.0	1.0	1.0
$n_{d,\max}(-)$	2.0	2.0	2.0
$R_{s,\min}(\Omega)$	0.001	0.001	0.001
$R_{s,\max}(\Omega)$	1.0	1.0	2.0
$R_{sh,\min}(\Omega)$	50	50	50
$R_{sh,\max}(\Omega)$	$1,100$	550	$1,500$

transcendental equation (4.6b) for I_{pv} with respect to all possible values of V_{pv} with a Newton-Raphson based method. The procedure needs to calculate a complete $I - V$ curve to be able to extract MPP and its execution time depends significantly on the step-size of V_{pv} variation.

From Figure 4.2, it can be seen that the generated power by a PV module in terms of PV voltage (P-V curve) is a concave function. Consequently, objective function of the problem in (4.6) is convex over feasible values of V_{pv} and I_{pv} and it can be solved employing well-known non-linear convex optimization techniques. In this study, IPM has been selected chiefly because it solves the problem fast. Furthermore, there exists a powerful still freely accessible implementation of IPM, i.e., IPOPT [125].

The fundamental idea of IPM is to transform the non-linear optimization problem with inequality constraints to a problem with only equality constraints. Then the optimum value is estimated through satisfying the necessary conditions proposed by Karush-Kuhn-Tucker (KKT) theorem.

Table 4.4 summarizes execution time of the curve-inspecting and the proposed methods to solve the problem given in (4.6). It illustrates that the curve-inspecting technique is computationally quite slow. Table 4.5 illustrate a summary of MATLAB scripts developed for the proposed MPP estimation algorithm. While the implementation of the proposed MPP estimation algorithm in MATLAB is more than 10 times faster, employing IPOPT makes it even faster. MPP estimation is a critical task of the PV module identification problem because it is executed several times depending on the identification algorithm and its parameters, e.g., population size and the maximum number of generations in a GA. A fast MPP estimation algorithm, in particular, speeds up substantially the identification algorithm.

Table 4.4: Execution times for different MPP estimation methods.

Methods	**Eexecution time**
Curve-inspecting method with MATLAB	$3000(msec)$
The proposed algorithm with MATLAB	$300(msec)$
The proposed algorithm with IPOPT	$90(msec)$

Table 4.5: MATLAB scripts of the proposed MPP estimation algorithm.

```
%MATLAB script of the MPP estimation algorithm. The objective function and constraints have
been attached for simplicity's sake. They need to be separate files with the same names. STC
or NOCT parameters need to be configured first.

iph = iph_STC;
a_Tc = a_STC;
i0 = i0_STC;

% A solver for the optimization problem
x0 = [voc_STC * 0.8, isc_STC * 0.7];      %initial guess
opt_lb = [voc_STC * 0.5, isc_STC * 0.5]; %lower bounds
opt_ub = [voc_STC, isc_STC];              %upper bounds
options = optimset('Algorithm','interior-point', 'TolFun', 1e-18, 'Display', 'off');
[ x,fval ] = fmincon(@PVObj,x0,[],[],[],[],opt_lb ...
                opt_ub,@(x)PVConstraints(x, var3, iph_STC,...
                        i0_STC, var1, var2, a_STC/var3),options);
[ c, ceq ] = confun(x);
vmpp = x(2);
impp = x(1);
pmpp = vmpp * impp;

% objective function
function f = PVObj(x)
    f = 1/(x(1) * x(2));
end

% constraints function
function [ c, ceq ] = PVConstraints(x, nd, iph, i0, rs, rsh, vt)
    % Non-linear equality constraints
    c = [];
    % inequality constraints
    ceq = [(x(1) - iph + i0 * (exp(( rs * x(1) + x(2)) / (nd * vt)) - 1) +...
            ((rs * x(1) + x(2)) / rsh)) ] ;
end
```

4.4.1 Model validation

The proposed MPP estimation algorithm is compared with the results of the curve-investigating method [46], i.e. a method to find MPP with any required accuracy employing an exhaustive search. Table 4.6 shows the results of this comparison. In Table 4.6, it can be seen that they are matched together precisely and the proposed algorithm estimates MPPs quite accurately comparing with experimental values given in Table 4.1.

Table 4.6: A comparison between the proposed MPP estimation algorithm and curve-investigating method [46].

Methods	KC200GT		ST40		E20/333	
	STC	NOCT	STC	NOCT	STC	NOCT
Curve-investigating method: $\begin{pmatrix} V_{mpp} \\ I_{mpp} \end{pmatrix}$	$\begin{pmatrix} 26.3000 \\ 7.6100 \end{pmatrix}$	$\begin{pmatrix} 23.2000 \\ 6.1300 \end{pmatrix}$	$\begin{pmatrix} 16.5980 \\ 2.4100 \end{pmatrix}$	$\begin{pmatrix} 14.7030 \\ 1.8840 \end{pmatrix}$	$\begin{pmatrix} 54.7000 \\ 6.0900 \end{pmatrix}$	$\begin{pmatrix} 50.4000 \\ 4.9100 \end{pmatrix}$
The proposed method: $\begin{pmatrix} V_{mpp} \\ I_{mpp} \end{pmatrix}$	$\begin{pmatrix} 26.3001 \\ 7.6099 \end{pmatrix}$	$\begin{pmatrix} 23.1994 \\ 6.1300 \end{pmatrix}$	$\begin{pmatrix} 16.5990 \\ 2.4099 \end{pmatrix}$	$\begin{pmatrix} 14.7000 \\ 1.8844 \end{pmatrix}$	$\begin{pmatrix} 54.7002 \\ 6.0899 \end{pmatrix}$	$\begin{pmatrix} 50.4002 \\ 4.9099 \end{pmatrix}$

4.5 PV model identification algorithm

The well-known non-linear programming (NLP) algorithms such as IPM cannot solve the PV model identification problem in (4.12) because it is a global optimization problem with multiple local optima. To address this problem, a GA is proposed in this chapter to solve the resulting global optimisation.

4.5.1 Genetic algorithms

Generally speaking, a GA is a search technique that mimics the natural genetic-based biological organisms. It is a class of stochastic search methods combining elements of two strategies, namely, exploring the search space and exploiting the best solution [126]. GAs also belong to a larger set of evolutionary algorithms. Due to similarity with Darwinian evolution, biological terms are widely used to describe GAs.

Unlike the gradient-based algorithms that looking for an individual solution, GAs deal with a population, as a set of solutions. GAs start with an initial fixed-size population, which is a set of randomly generated *chromosomes*. A chromosome concatenates the numerical representations of all decision variables. Theoretically, these variables can be of any types of information; however, most of the prior developed GAs are focused on binary or real values. Having discrete values like integers, the resulting optimisation problem is of the class of mixed integer problems which requires more complications to be considered.

GAs rely on three stochastic operators, namely, *crossover*, *mutation*, and *selection* [127]. A GA generates new offsprings or solutions by employing either a crossover or a mutation operation. The crossover operators take two mated chromosomes of

the current generation, as parents, and generate two offsprings by randomly mixing the parents chromosomes. Moreover, to avoid quick convergence to the local optima points, it is essential to introduce new solutions to algorithm by randomly changing one or more decision variables of a part of chromosomes. All chromosomes, including the original and new generated ones, are evaluated according to the defined fitness function. Then, the selection operator selects the predefined population size number of the most fittest chromosomes to construct a new generation. If the problem is well-posed, chromosomes iteratively converge to the global optimal solution through a number of generations and according to a selection rule states that the fitter chromosomes have higher probabilities of being selected.

For further details on GAs, the reader is referred to [126].

4.5.2 Design of the proposed PV model identification algorithm

Table 4.7 presents a pseudocode of the proposed GA in this study. The proposed GA adopts disruptive selection method [128] to find the global optima. Traditional GAs allocate more trials to above-average chromosomes and do not guarantee convergence to the global optimum. Unlike traditional GAs, the disruptive selection, which is a more effective method, gives a higher probability to both the best and the worst chromosomes (Figure 4.5) in a roulette wheel framework [128]. Figure 4.5 (a) and (b) illustrate MPPs proportional to chromosomes of the last generation and indicate varieties of genes. Figure 4.5 (c) shows the effect of the disruptive selection on the fitness values. While it preserves the best chromosomes through generations, it still keeps the average fitness of the population around half of the best fitness.

In order to solve the proposed global optimization problem having multiple local optima, it is essential to employ methods that prevent the population premature convergence. The proposed GA employs two techniques to preserve population diversity through generations. Firstly, a specific technique is used to adjust the mutation rate dynamically. It changes the mutation rate proportional to a diversity measurement, which is defined as the ratio of the average fitness to the best fitness [126]. Secondly, random offspring generation is the other technique used to preserve the population diversity. When two parents have equal genotypes, normal offsprings are clone to their parents. The idea behind this method is that if this situation happens, then the

Table 4.7: A pseudocode of the proposed PV model identification method.

Step1: Initialization

Load optimization parameters: popSize, genNo_max, mutRate_init, crossoverRate;

Load PV parameters from datasheets;

$genIndx = 0$; current generation

$mutRate\_max = min\{3 \times mutRate\_init, 0.9\}$;

$v = [n_d, R_s, R_{sh}]$; chromosme structure

$pop = popGen(popSize)$; Random initial population of popSize

Step 2: Crossover and mutation

Selecting parents ($\lambda = rand() \in [0,1]; i \in \{1,2\}$):

$[parent_{1,c}, parent_{2,c}] = RouletteWheel(crossoverRate)$;

$offspring_{i,c} = \lambda parent_{1,c} + (1-\lambda) parent_{2,c}$;

$mutRate = mutRate\_init + (mutRate\_max - mutRate\_init)(\frac{\overline{fitness_{genIndx}}}{fitness_{best,genIndx}})^a$;

Selecting chromosomes/variable to be mutated:

$[chrom, var]_m = RouletteWheel(mutRate)$;

$mutVar = var_m \times rand() \times (1 - \frac{genIndx}{genNo\_max})$;

$migratedChrom_m = chrom_m(mutVar)$; using mutated var.

$pop = pop \cup \{offspring_{i,c}\} \cup \{migratedChrom_m\}$;

Step 3: Calculating fitness values (j is the chromosome number)

Calculating values of objective functions:;

$(v_{mpp,stc}, i_{mpp,stc})$ = Solve Eq. (4.6) for STC with IPM;

$(v_{mpp,noct}, i_{mpp,noct})$ = Solve Eq. (4.6) for NOCT with IPM;

$[f_{1,j}, f_{2,j}]$ = Use these values acc. to Eqs. (4.7) and (4.8);

$normalizeFactor_1 = \underset{j \in [0, popSize]}{\text{maximize}} \{f_{1,j}\}$;

$normalizeFactor_2 = \underset{j \in [0, popSize]}{\text{maximize}} \{f_{2,j}\}$;

$fitness_j = \frac{0.5}{normalizedFactor_1} f_{1,j} + \frac{0.5}{normalizedFactor_2} f_{2,j}$; Eqs. (4.9), (4.11b) and (4.8)

Step 4: Selection

$P_{sel,j} = \frac{|fitness_j - \overline{fittness_{genIndx}}|}{fitness_{best,genIndx}}$; selection probability

$pop = RouletteWheel(P_{sel,j})$; new population of popSize

Step 5: Terminating criteria

If $genIndx < genNo\_max$ AND $fitness_{best,genIndx} < 10000$ then

update genInx and go to step 2

else chromosome $[n_d, R_s, R_{sh}]$ with $fitness_{best,genIndx}$

is the optimum solution. STOP.

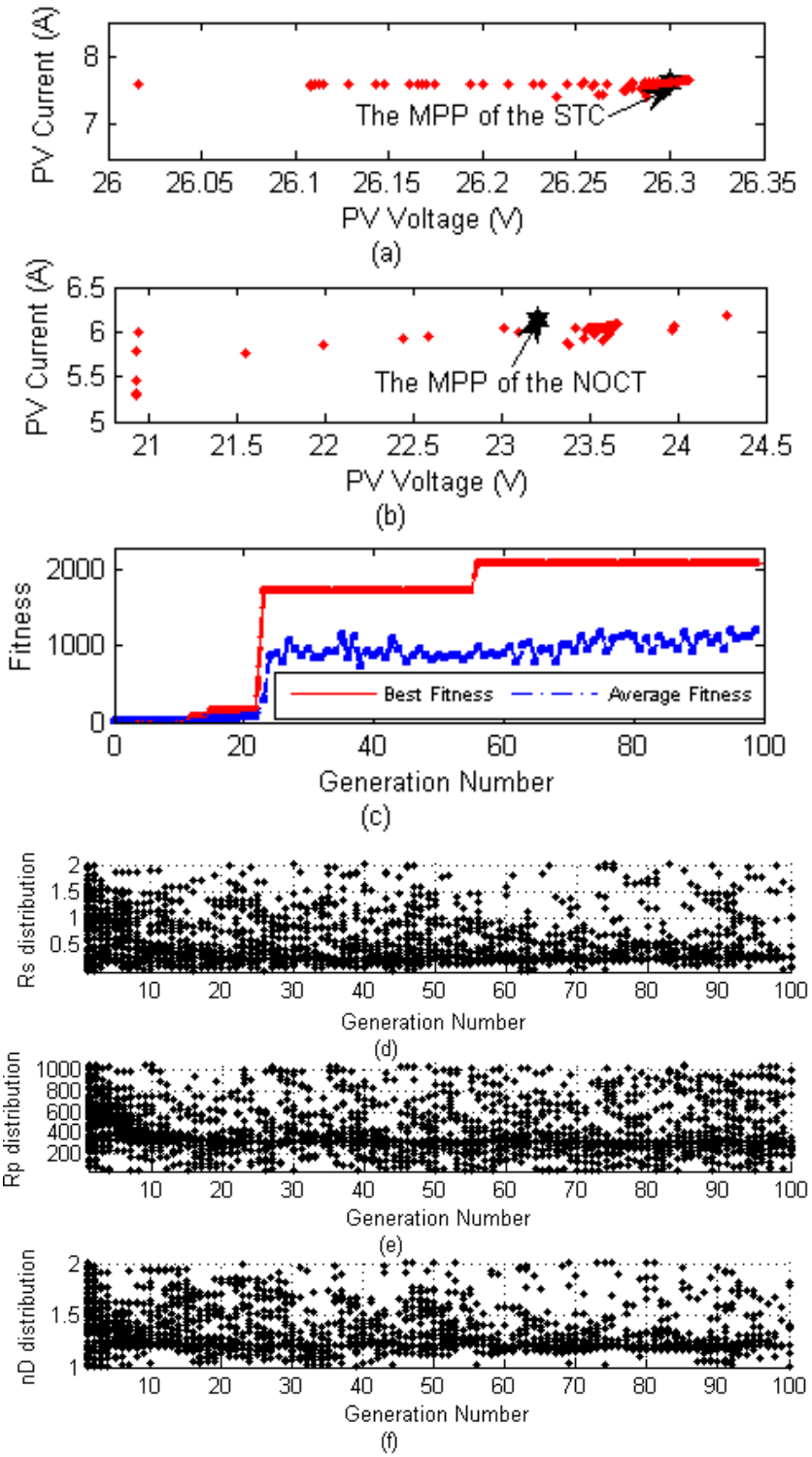


Figure 4.5: Generated curves for KC200GT PV module employing the proposed STC-based method: (a) estimated MPPs of STC during the last generation, (b) estimated MPPs of NOCT during the last generation, (c) the best and average fitness values. (d)-(f) Diversities of chromosomes for all generations.

normal offsprings are replaced by new randomly generated ones [129]. In Figure 4.5 (c), it can be seen that although the best fitness is constant for several consecutive generations, which means the algorithm is vulnerable to premature convergence, the average fitness is variable. In other words, the proposed GA preserves the population diversity through generations. Figure 4.5 (d)-(f) depict diversities of each parameter in more details.

From Table 4.7 , it can be seen that the proposed procedure calculates the maximum values of each objective function prior to selecting chromosomes of the next population. These values are the normalization factors $\{\theta_i\}$ in (4.12) for each objective function.

Unlike the identification methods in [46] and [108], the proposed method does not require an initial guess for ideality factor. Moreover, execution time is independent of the parameters resolution and the method can be terminated in the middle of execution according to a accuracy-time trade-off that leads to have sub-optimal solutions.

4.5.3 Model validation

The proposed PV identification method is used to identify equivalent electrical circuit of three PV modules with different technologies [130–132]. Table 4.1 illustrates the data of these PV modules provided by three manufacturers: Kyocera, Shell, and Sunpower. Subsection 4.5.4 presents identified model for Sunpower E20/333 module using different variations of the proposed PV model identification method. It also validates the results of the proposed method by comparing them with experimental curves, available in PV module datasheet [132]. The results are also compared in Subsection 4.5.5 with two other studies in [46] and [92], indicating substantial improvements in accuracy. All implementations are conducted using MATLAB with the following parameters:

- Design variables: $[n_d, R_s, R_{sh}]$
- Crossover rate: 85%
- Initial mutation rate: 40%
- Population size : 100

- The maximum number of generations: 100
- Convex weights: $[0.5, 0.5]$

4.5.4 Sunpower E20/333 model identification and experimental validation

Table 4.8 presents electrical parameters of Sunpower E20/333 PV module identified by different variations of the proposed method. It shows a comparison between different errors in STC-based, NOCT-based, and full methods indicating how the latter provides a solution with the smallest overall error, with respect to experimental MPPs (Table 4.1), than the other variations (with 5.1% improvement). The identified solution by the full method is close to the results of the STC-based method, however, it is not necessarily the same for other types of PV modules. From Table 4.9, it can be seen for ST40 module that the identified solution by the full method is much closer to the results of NOCT-based method and it experiences an improvement of 31.5% compared with the STC-based method.

Figure 4.6 compares the predicted $I - V$ curves of E20/333 PV module, for different amounts of solar irradiance and cell temperature, with respect to the benchmark points provided by manufacturer (the circle markers). It is observed that the identified model by the proposed full method predicts different curves very close to the empirical benchmark points.

Figure 4.7 depicts the *curve error* versus PV voltage for different amounts of solar irradiance. The curve error is defined as absolute normalized difference between the predicted $I - V$ curve, which is generated by employing the identified model, and experimental benchmark $I - V$ curves which is provided by manufacturers. If there exist predicted and empirical points with the same value of voltage (or current) on curves, then the error equals to absolute difference between the proportional current (or voltage) values. The calculated absolute values are normalized with either short-circuit current or open-circuit voltage values.

4.5.5 Accuracy verification of the proposed method

From Figures 4.6 and 4.7 and explanations under subsections 4.5.4, it can be seen that the proposed method extracts a PV model that predicts accurately matched $I - V$ curves to experimental measurements, provided by manufacturers. In order

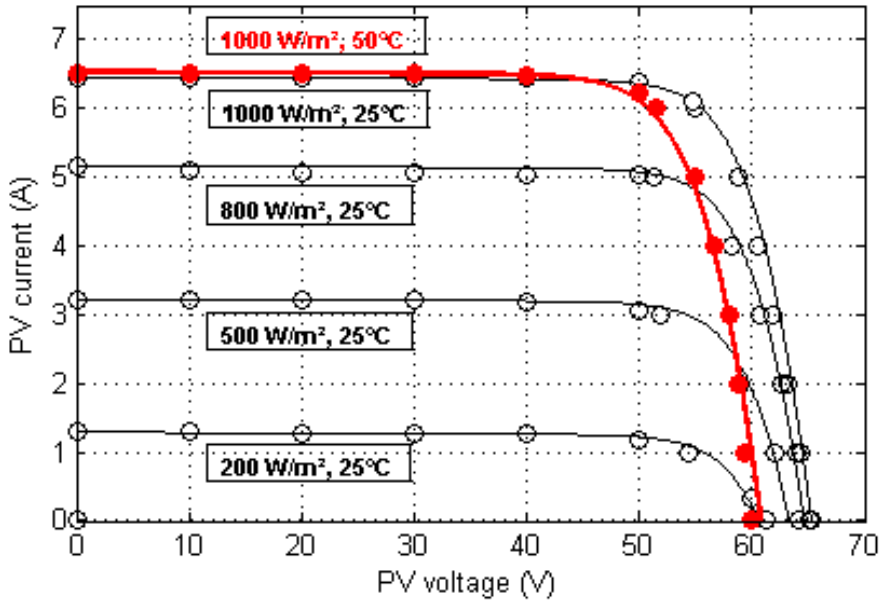


Figure 4.6: Predicted $I-V$ curves (the solid lines) and experimental points (the circle markers) of Sunpower $E20/333$ PV module for different values of solar irradiance and $T_c = 25°C$, in addition to the case that $T_c = 50°C$ (the thicker line and the solid circle markers).

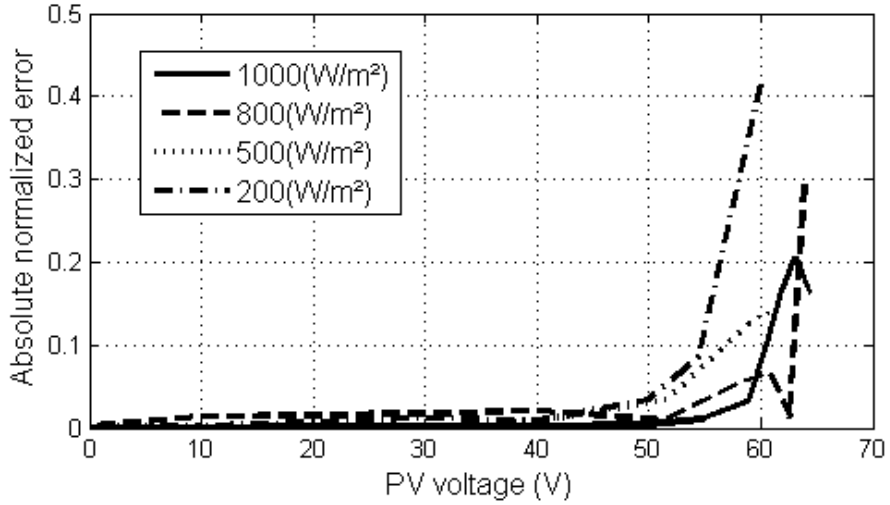


Figure 4.7: Absolute normalized curve errors between the experimental and predicted $I-V$ curves of E20/333 module at different amounts of solar irradiance.

to indicate improved accuracy of the proposed method, the measured identification error, as the sum of STC- and NOCT-identification errors, is also compared with the corresponding values introduced by the methods in [46] and [92]. Figure 4.8 shows this comparison for different PV modules. It is observed that the identified model by the proposed full method generates the least overall error among all methods. For instance, applying the proposed full method to KC200GT and ST40 modules and comparing with the results in [46] and [92] delivers a significant improvement. Identically, substantial improvements with respect to these references are gained for

Table 4.8: Identified electrical parameters of Sunpower E20333 (Monocrystalline) module employing the proposed methods.

Parameters	STC-based	NOCT-based	Full
$n_d(-)$	1.132	1.014	1.167
$R_s(\Omega)$	0.365	0.598	0.337
$R_{sh}(\Omega)$	1125.75	1399.17	1414.89
STC-identification error$(\%)$	0.0004	1.6279	0.0121
NOCT-identification error$(\%)$	1.0927	0.0014	1.0208
Overall error$(\%)$	**1.0931**	**1.6293**	**1.0375**

Table 4.9: Identified electrical parameters of SHELL ST40 (Thin film) and Kyocera KC200GT (Polycrystalline) modules employing the proposed methods.

	SHELL ST40			Kyocera KC200GT		
Parameters	**STC-based**	**NOCT-based**	**Full**	**STC-based**	**NOCT-based**	**Full**
$n_d(-)$	1.346	1.829	1.973	1.291	1.033	1.348
$R_s(\Omega)$	1.508	1.151	1.067	0.232	0.421	0.214
$R_{sh}(\Omega)$	386.22	294.59	510.49	547.91	662.813	1060.66
STC-identification error$(\%)$	0.0087	3.2138	3.0240	0.0007	3.1924	0.0291
NOCT-identification error$(\%)$	4.4390	0.0020	0.0210	2.190	0.0030	2.0960
Overall error$(\%)$	**4.4477**	**3.2158**	**3.0450**	**2.191**	**3.1958**	**2.1254**

STC-identification errors when STC-based method is applied to these two modules.

The absolute normalized curve error between predicted and experimental $I-V$ curves of ST40 module for two different values of insolation are given in Figure 4.9. The absolute errors have been normalized using similar technique introduced under Figure 4.7. From Figure 4.9, it can be seen that the proposed full method shows a significant reduction in curve errors. Moreover, it gives dramatic improvements especially for lower insolation.

The introduced improvement by applying the proposed full method may vary for different types of PV modules; however, the full method is still a consistent identification algorithm. It means that it always proposes the optimum solution with respect to the overall and curve errors, regardless of different PV technologies. A PV module is dominated by either NOCT (e.g. ST40 module) or STC condition (e.g. KC200GT or E20/333 modules). If it is dominated by NOCT condition, using the STC-based identification method (e.g. the methods in [46] and [92]) provides a model with sig-

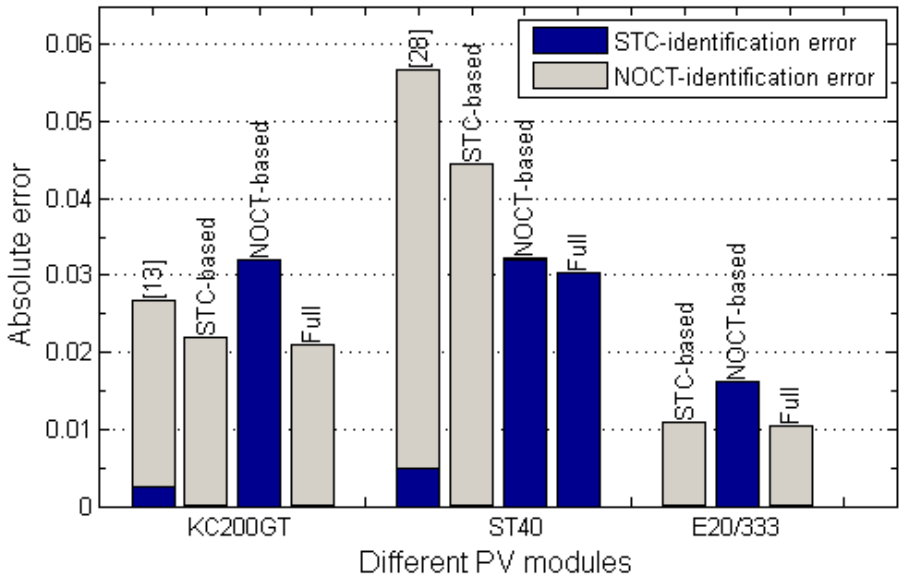


Figure 4.8: Identification error of the proposed methods comparing with the results in [46] and [92].

nificant overall error chiefly because of a substantial NOCT-identification error. The proposed full method is consistent, i.e., it automatically adjusts itself to find the best equivalent electrical model for a PV module, regardless of being dominated by either STC or NOCT condition.

4.6 Summary

This chapter proposed new formulations for PV model identification and MPP estimation problems. These formulations construct proportional non-linear optimization problems using only the main algebraic equation of PV modules and the available data provided by manufacturers. A novel GA- and IPM-based identification method was introduced to solve the resulting multi-objective global optimization problem, which has several local optima. The proposed method identifies parameters of an equivalent electrical circuit as a model to predict the behavior of PV modules at any operating condition. The main aspect of the proposed model identification method is to extract these electrical parameters with minimizing the overall identification error rather than just STC-identification error. While the latter is the distance between empirical and estimated MPPs of STC curve, the overall identification error is the sum of such distances for both STC and NOCT curves. The proposed MPP estimation algorithm is based on IPM and finds MPPs much faster comparing with two recent approaches. Unlike other approaches that blindly search $I - V$ curve, this algo-

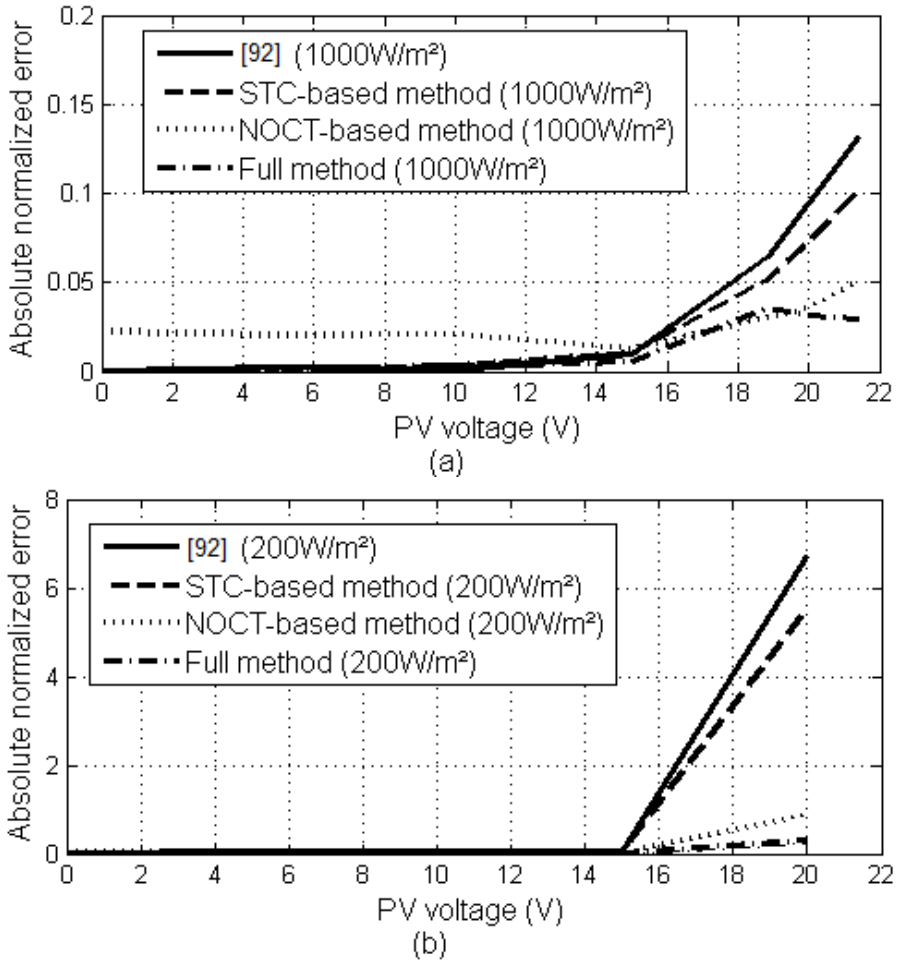


Figure 4.9: Absolute normalized curve errors between experimental and predicted $I-V$ curves of ST40 module introduced by different methods at (a) $1000W/m^2$ and (b) $200W/m^2$ insolation.

rithm quickly solves an optimization problem to extract MPP. Applying the proposed identification method to three PV modules of different technologies shows very low identification errors. The results were compared with empirical curves as well as two recent researches. It is indicated that the proposed method works well even for PV modules dominated by NOCT condition. A pseudocode of the proposed method and a sample MATLAB script of the MPP estimation algorithm were presented.

Chapter 5

Dynamical Simulation of Standalone dc Microgrids

In order to be able to analyze different control strategies, it is essential to simulate standalone dc microgrids operations. Moreover, long-term simulations of standalone dc microgrids, in particular, is necessary to predict their behaviours under wind speed and solar irradiance uncertainties. This chapter provides the simulation results of a sample standalone dc microgrid which cover both the fast and slow dynamics of the system. It is shown that such a standalone dc microgrid is simulated by employing acausal approach, using flat hybrid DAEs. A MATLAB/SimPowerSystems acausal model is developed in order to simulate the system including all fast and slow dynamics. On the other hand, the developed Modelica model in Chapter 2 is used for long-term simulation with the cost of ignoring fast dynamics.

5.1 Introduction

In order to study the behavior of standalone microgrids, it is essential to simulate the system. Such simulations are necessary regarding two aspects: i) fast dynamics observations leading to short-term simulations; and ii) long-term simulation. A short-term simulation is required to be able to design and analyze inner controllers. A long-term simulation, on the other hand, is essential to predict the standalone microgrids behaviours over a period of time ahead. Such a long-term simulation is fundamental to design and analyze outer level model predictive controllers as well as to handle

different uncertainties in smart grid applications. The outer control strategies and long-term simulations operate in the order of seconds, minutes or even hours for which the fast dynamics can be replaced with the relevant steady-state equations.

Both the short- and long-term simulations of standalone microgrids are challenging due to the three major issues. Firstly, PV module, wind turbine, and battery bank add algebraic constraints to the system model. Such embedded algebraic constraints shifts the problem to the class of DAEs. Secondly, the multi-mode operation of battery bank introduces discontinuity and makes the system to be of the class of hybrid DAEs. Finally, due to existence of simultaneous fast and slow dynamics, microgrids simulation is becoming time-consuming and memory expensive. For these reasons, standalone microgrids should be simulated employing an acausal approach.

As explained in Chapter 1, acausal system modelling is a declarative approach in which individual parts of the system are directly described as HDAE equations [22]. Among the available SIMULINK toolboxes, SimPowerSystems toolbox employs the acausal approach to model and simulate the system. Moreover, Modelica-based simulation frameworks use acausal approach to simulate complex system. There are a number of available Modelica modelling and simulation frameworks, among them the open-source OpenModelica environment [28] is used throughout this research, chiefly because it is a free software and it supports more features of Modelica language comparing to others. OpenModelica provides and integrated Modelica compiler and DASSL [137] general purpose hybrid DAE solver.

There are prior researches focused on using either basic SIMULINK or SimPowerSystems toolboxes of MATLAB/SIMULINK framework to simulate different components of sustainable microgrids. A simplified SIMULINK model of PV array is presented in [133]. Such a model is also employed in [133] to simulate PV arrays that supply constant ac and dc loads at specific operating conditions. They used the linearization technique introduced in [134] to take into account the effect of solar irradiation on cell temperature. The work in [46], on the other hand, proposes two simulation scenarios based on equivalent electrical circuits of PV modules. The proposed equivalent electrical circuits are based on the SimPowerSystems components and consists of controlled current sources. In order to increase the simulation accuracy at low solar radiation, a two-diode PV model is introduced in [135].

A mathematical representation of a 3-blade wind turbine operation is presented in [94]. Based on such a mathematical representation, SimPowerSystems toolbox provides an integrated ready-to-use component to simulate 3-blade wind turbines. Moreover, the works in [136] and [19], respectively, propose comprehensive mathematical models of PMSGs and lead-acid batteries which are used by SimPowerSystems in its ready-to-use simulation components.

5.2 Standalone microgrids simulation using SimPowerSystems toolbox

Figure 5.1 illustrates the developed MATLAB/SIMULINK model of a sample standalone dc microgrid. From Figure 5.1, it can be seen that the system decomposed into ten components, including wind speed and solar irradiance generators, PV array, wind turbine, PMSG, rectifier, battery bank, dc load, and a boost and a buck converter to connect, respectively, solar and wind branches to a dc bus. The resulting model is simulated according to the acausal approach available by SimPowerSystems framework.

5.2.1 Solar irradiance and wind speed generators

Although solar irradiance and wind speed levels can also be given deterministically with their profiles, there are embedded modules to stochastically simulate these values using Beta and Weibull random generators. These generators use hourly shape parameters which are extracted and analyzed in Chapter 3.

As the main advantage, stochastic models of wind speed and solar irradiance take uncertainties into account for simulation purposes. For instance, these random values for solar irradiance inherently take into account uncertainties due to cloud shading or incident angle and generated values for solar irradiance can be directly fed into a PV array. Stochastic models are also simple to implement. Once all hourly shape parameters are learned, wind speed and insolation values can be generated by using random generators.

Figure 5.2 shows the developed simulation model of the hourly solar irradiance generator which is also applicable to a wind speed generator. While shape parameters and extraterrestrial insolation for different locations is applied through two-

dimensional lookup tables, a random generator uses these parameters to generate hourly clearness index that can be multiplied by extraterrestrial insolation of that day to calculate hourly global solar irradiance. The result is segmented and is step-wise linearized using a linear interpolator. Figure 5.3 depicts the sample simulation results ontained for hourly insolation in Dyfed, UK, on 01 January and 01 October.

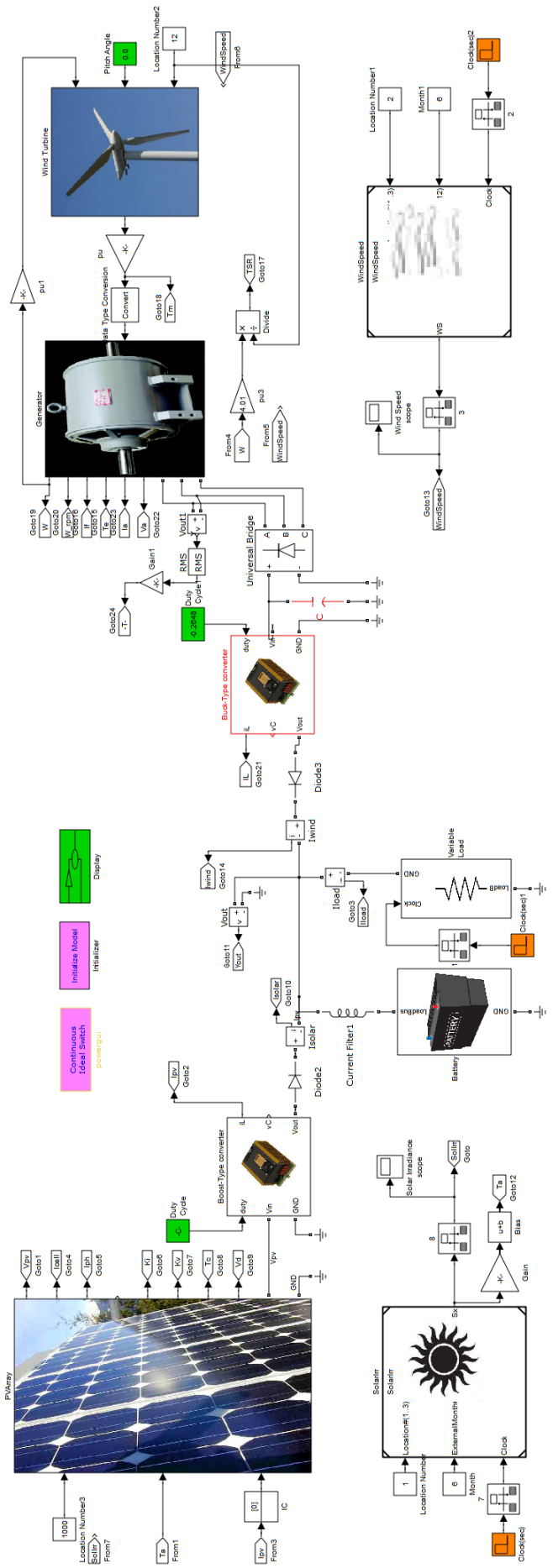


Figure 5.1: SIMULINK model of a hybrid wind-solar-battery renewable energy system.

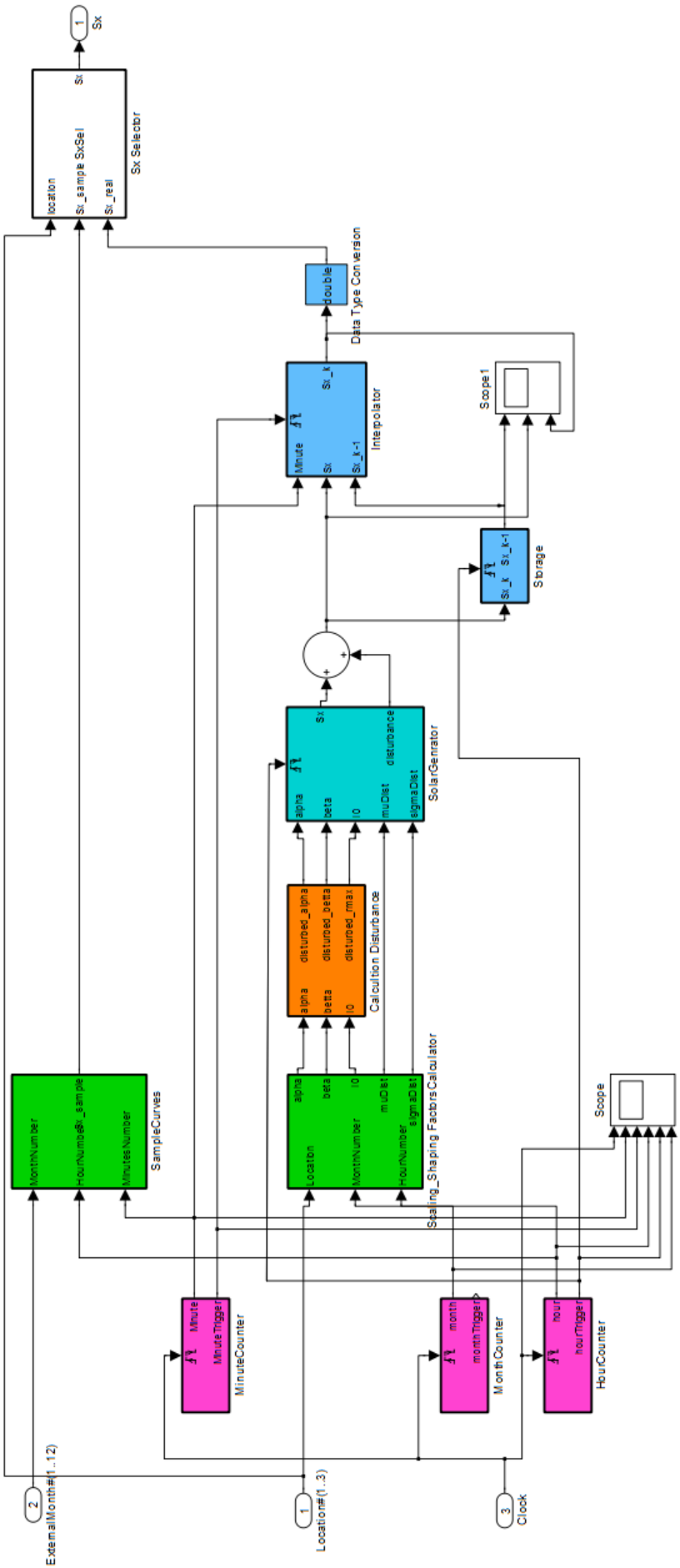
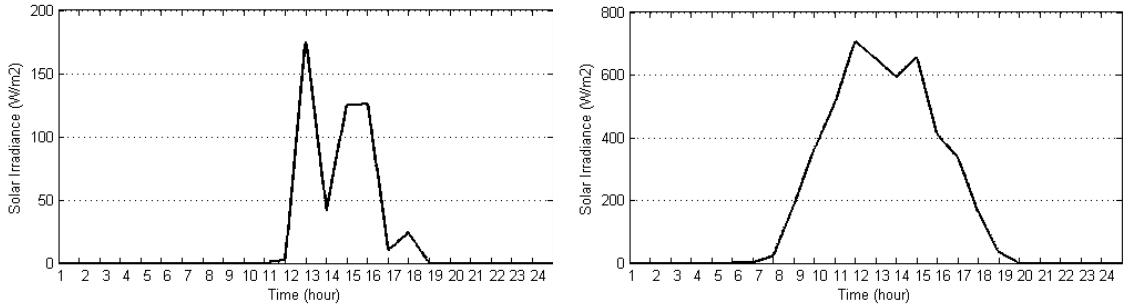


Figure 5.2: Solar irradiance generator module.



(a) (b)

Figure 5.3: Simulated hourly solar irradiance in Dyfed, UK, on (a) 01 January; and (b) 01 October.

Coupling wind speed and insolation generators with other components causes the simulation to be multi-rate. While the time step for all components is small enough to simulate system dynamics, the simulation rate of wind speed and insolation is one hour, which is significantly higher. In this study, this challenge has been overcome by feeding a separate clock to the wind speed and solar irradiance generators (Figure 5.1).

5.2.2 PV array simulation

Figure 5.4 illustrates the block diagram of a PV array simulator. In Figure 5.4, it can be seen that this electrical circuit includes a current source which is controlled by (5.1), where I_m is the sum of photocurrent, I_{ph}, and diode current, I_d. PV cell temperature, T_c is simulated externally using a state-space block and its output is then fed into the PV module simulator.

$$I_m(t) = I_{ph}(t) - I_0(t)\left\{\exp\left(\frac{V_{pv}(t) + R_s I_{pv}(t)}{n_d N_s}\frac{q}{KT_c(t)}\right) - 1\right\}. \tag{5.1}$$

Although a PV array as a component can be simulated with its equivalent electrical circuit, there is still a challenge to connect it to a DC-DC converter. Since a PV array is an instantaneous system, its connection to a converter casts an algebraic constraint on the converter which is not solvable with ordinary ODE integrators. SIMULINK employs a proprietary index-1 DAE solver consists of an ODE integrator coupled with a Newton type method to solve algebraic constraints. It requires using a small step-size for simulation (i.e. 10^-8 sec. in this study), as well as setting manually a consistent initial value.

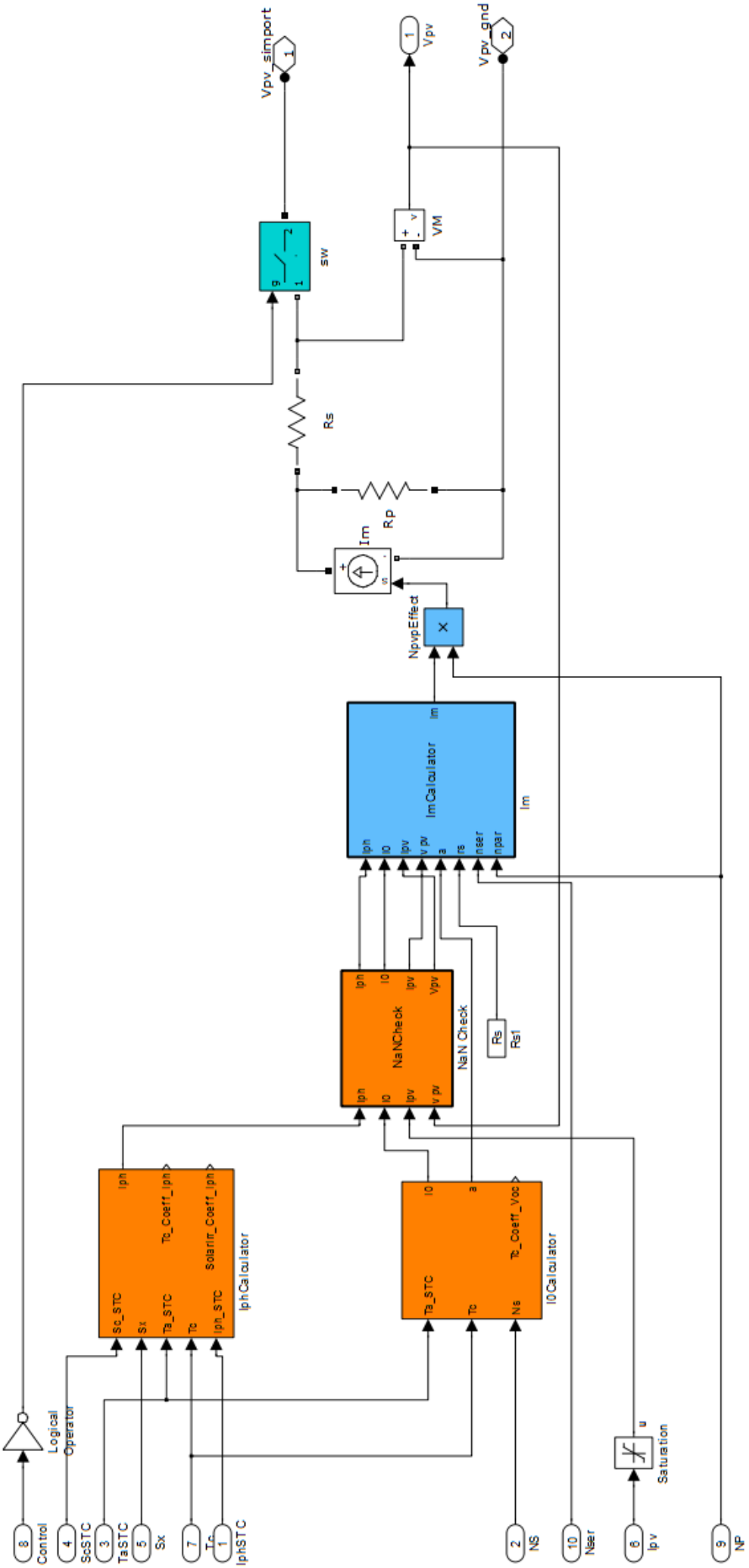


Figure 5.4: PV module simulation model.

5.2.3 Boost- and buck-type DC-DC converter simulation

Figure 5.5 shows a SimPowerSystems circuit, including pulse width modulation (PWM) block and ideal switches, which is developed to simulate a boost-type DC-DC converter. Buck-type converters are are also modelled as electrical circuits. From Figure 5.5, it can be seen that the simulation of converters is not simplified with averaging approach, instead it is modelled as a hybrid system. Transistors are replaced with two ideal switches for simplicity. In order to prevent cases in which both switches are opened causing the inductor to be disconnected, a snubber resistor is also added to Ideal Switch1.

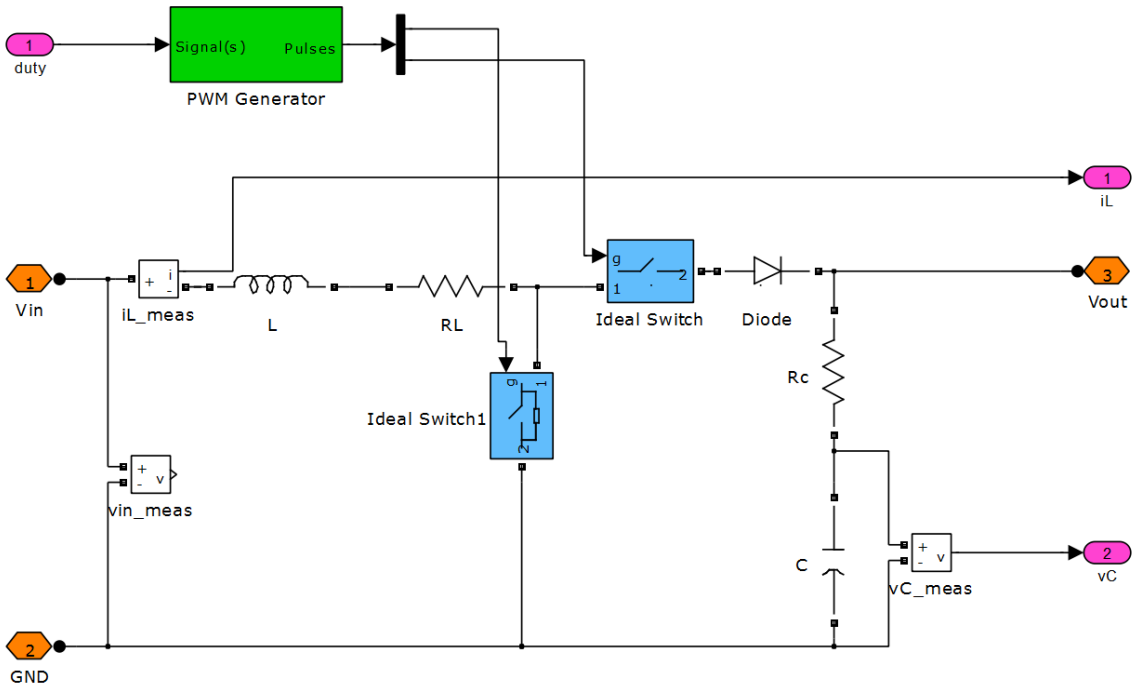


Figure 5.5: Boost-type dc-dc converter simulation model with ideal switches and inductor resistive loss.

5.2.4 Battery simulation

SIMULINK provides a built-in module to simulate different types of batteries, including lead-acid battery. The provided module is based on the mathematical model for lead-acid battery presented in Chapter 2.

Connecting a battery bank to the system, similar to PV arrays, introduces algebraic constraints. Consequently, in order to simulate the resulting DAE system, step-size of simulation is reduced and a slight in series inductance is added (Figure 5.1).

5.2.5 Wind branch simulation

SIMULINK/SimPowerSystems provides built-in modules, including a 3-blade wind turbine, a 3-phase PMSG, and a 3-phase passive rectifier, in order to simulate wind branch. These modules are used to construct the wind branch of the sample standalone sustainable microgrid in this study.

Table 5.1 summarizes wind branch parameters and their values.

5.2.6 Simulation results

The available built-in algebraic loop solver of SIMULINK helps SimPowerSystems toolbox to integrate the model of standalone dc microgrid. However, as mentioned before, it also requires applying a small simulation step-size, i.e. 10^{-8} seconds, and initial condition blocks (ICs) across algebraic loops. Moreover, in order to simulate pulse width modulators (PWMs) and hence dc-dc converters, simulation step-size must be fixed. In other words, fixed-step solvers must be chosen instead of much faster variable-step solvers. Such small fixed simulation steps leads to a very slow and memory-expensive simulation which limits any possible duration of simulation to around 2 seconds, using AMD Athlon processor with $3.0\ GB$ of RAM.

TABLE 5.1 summarizes simulation parameters for wind, solar, and battery branches.

Figures 5.6 shows the simulation results obtained for solar branch of the sample microgrid. In Figures 5.6, it can be seen that generated power by solar branch of microgrid is less than the maximum power which is due to the chosen value for its duty cycle. Duty cycle is constantly kept at the value of 0.55 and hence the generated power of solar branch gradually reduced due to battery and dc bus voltage level increase. An external controller (see Chapter 6) adjusts solar branch duty cycle in order to ensure that PV array always operates at MPP, i.e. $(V_{mpp}, I_{mpp}, P_{mpp}) = (26.3\ V, 71.6\ A, 2.001\ KW\ )$ at STC.

Table 5.1: System parameters for SIMULINK/SimpowerSystems simulation.

Wind branch		Battery branch		Solar branch	
$P_{wt,nom}(KW)$	10.0	$V_{bstack,nom}(V)$	48.0	$Ppv_nom(KW)$	2.0
Turbine radius (m)	4.01	$C_{10}(Ah)$	550	Number of PV modules $(-)$	10
Duty cycle $(-)$	0.37			Duty cycle $(-)$	0.55
Wind speed (m/s)	12.0			Insolation (Wh/m^2)	1000.0
PWM Frequence (KHz)	20.0			PWM Frequence (KHz)	20.0
$\lambda_{nom}(-)$	7.35				

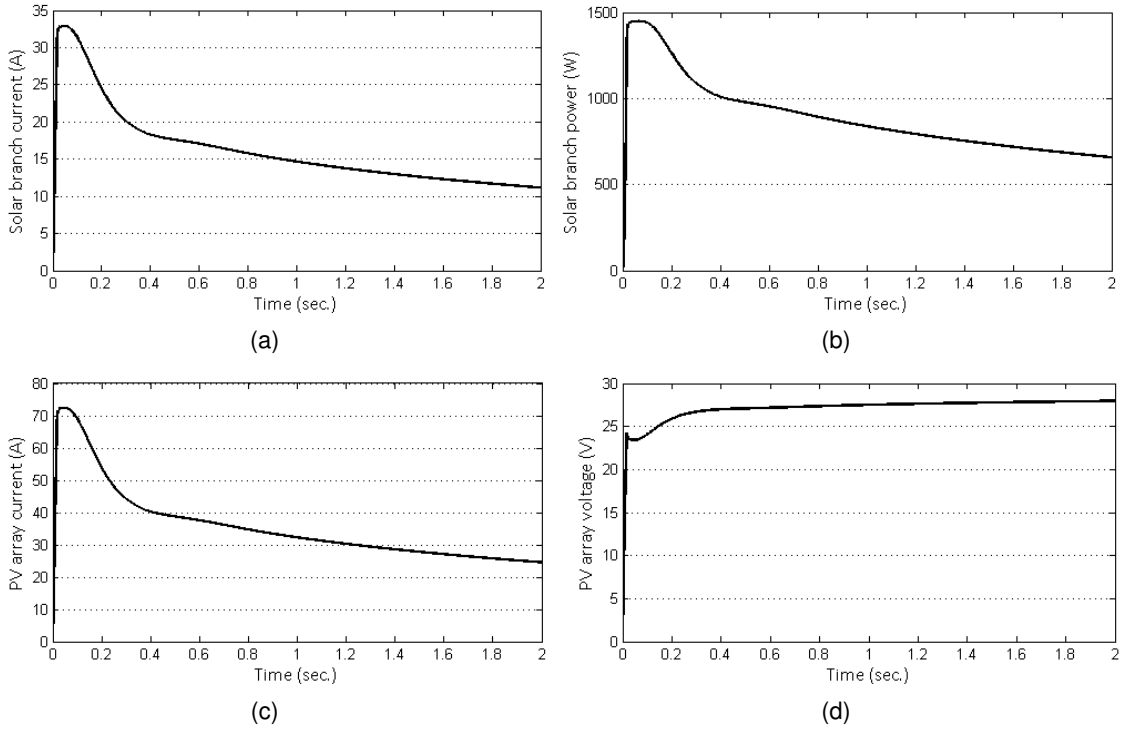


Figure 5.6: Simulated (a) current and (c) power of solar branch; and (b) current and (d) voltage of PV array using SimPowerSystems toolbox.

Figure 5.7 indicates that choosing a duty cycle value of 0.37 for wind branch, ensures that it operates in its MPP. From Figures 5.7a and 5.7b, respectively, it can be seen that wind branch generates $10\ KW$ of mechanical power at a rotational speed of $22\ rad/s$ which shows it operates in MPP. In the beginning, rotational speed of wind turbine is initialized to $10\ m/s$ that generates around $3\ KW$. As depicted in Figures 5.7c, there is a peak in applied mechanical torque to wind turbine to increase its rotational speed and hence generated power. However, after a transition the mechanical torque remains steady at $450\ N.m$ to supply the demanded electromagnetic torque at nominal rotational speed. From Figure 5.7d, it can be seen that there is no supplied electromagnetic torque during the time before $t = 0.2$ seconds that helps rotational speed to reach its nominal value. In Table 5.1, it can be seen that when wind speed is $12\ m/s$, wind turbine rotates with a tip speed ratio of 7.35 in order to generates the maximum power. As shown in Figure 5.7e, tip speed ratio remains steady at around 7.3 after a transition phase. Although wind turbine operates at its MPP and generates $10\ KW$, Figure 5.7f shows that the delivered power by wind branch is less due to the internal electrical loss of buck converter. Since the voltage level of wind branch equals to the battery bank voltage, i.e. around $60\ V$ at $t = 2\ sec$ (Figure 5.7g), the injected current by wind branch is around $135\ A$ (Figure 5.7h).

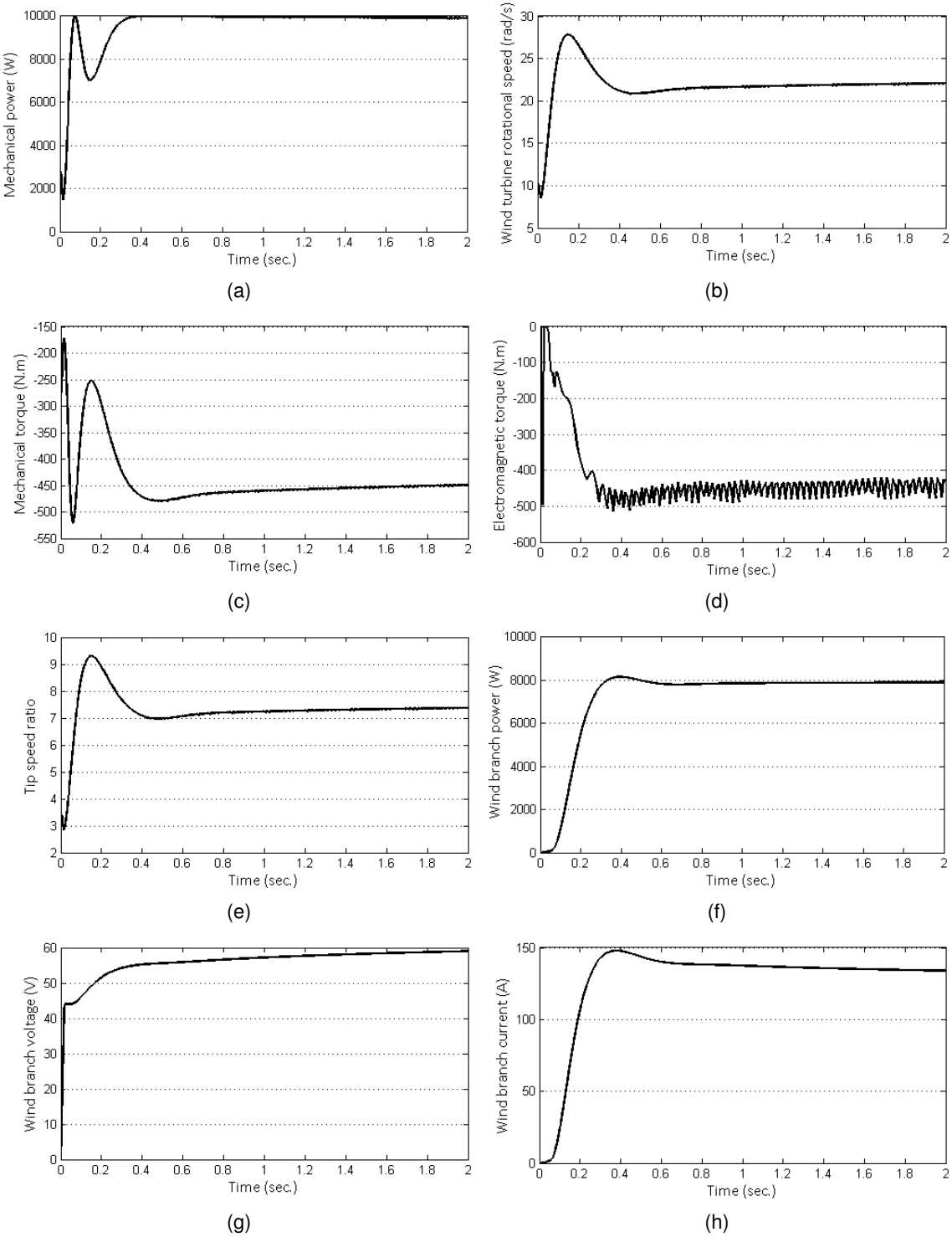


Figure 5.7: SIMULINK simulation results of wind branch including (a) mechanical power; (b) rotational speed; (c) mechanical and (d) electromagnetic torques; and (e) tip speed ratio variation. It also indicates generated (f) power and (g) current of wind branch injected to dc bus at the voltage level given in (h).

The simulation results obtained for battery bank are also depicted in Figure 5.8. From Figure 5.8a, it can be seen that in the beginning when wind branch does not operate at its MPP, battery bank is discharged in order to provide its share of load demands. Meanwhile, state of the charge (SOC) of battery bank goes down before $t = 0.2$ seconds. Then, battery bank starts being charged when the generated power by wind branch is increased. Figure 5.8b shows SOC variations of battery bank and indicates that it is in charging state after a short-time discharging. The similar state switching can also be observed from Figure 5.8c which depicts the battery bank current variations. It is important to note charging current and power are represented by negative values. From Figure 5.8d, it can be seen that the voltage level of battery bank is increased gradually during its charging state. In order to steadily generate the maximum solar- and wind-branch powers, an external controller must re-adjust duty cycles due to battery bank voltage variations.

Figure 5.9 depicts the electrical current and power delivered to the connected dc load. It is important to note that the dc bus voltage level and hence the delivered power to load are changed due to variations of the battery bank terminal voltage.

5.3 Modelica-based dynamical simulation of standalone microgrids

The proposed Modelica model in Chapter 2 is used to simulate a sample standalone dc microgrid, consisting of an array of Kyocera KC200GT PV modules, a 3-blade wind turbine connected to a PMSG, and a bank of Panasonic LC-R127R2PG lead-acid batteries. The works in [46] and [20] present the identified electrical parameters of the employed PV module, lead-acid battery, and wind turbine. Table 5.2 summarizes all parameters and their values used in this study.

Fig. 5.10a compares simulated current-voltage ($I - V$) curve of KC200GT PV module at STC condition against experimental curve available by the manufacturer in datasheet (the circle markers). It is observed that the proposed model predicts very close to the empirical data provided by manufacturer.

The simulation results obtained for a single LC-R127R2PG lead-acid battery over a full operating cycle, including charging, over-charging, saturation, discharging, over-discharging, and exhaustion zones, are given by Figures 5.10b and 5.10d. In Fig-

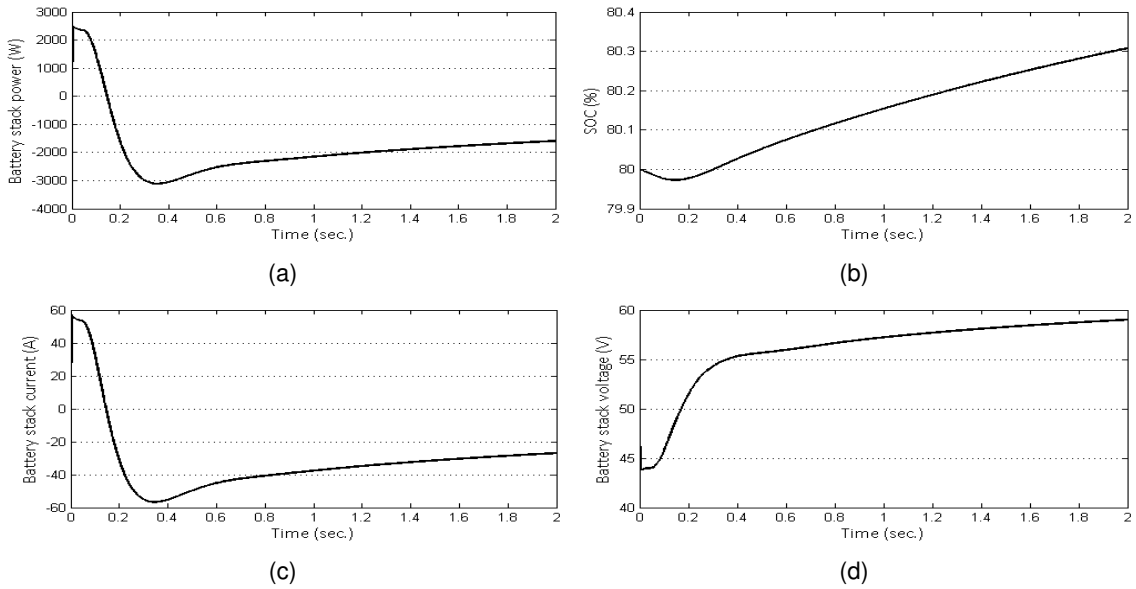


Figure 5.8: SIMULINK simulation results of battery bank: (a) charging (positive) and discharging (negative) power; (b) SOC variations; (c) charging (positive) and discharging (negative) current; and (d) battery bank voltage level variations.

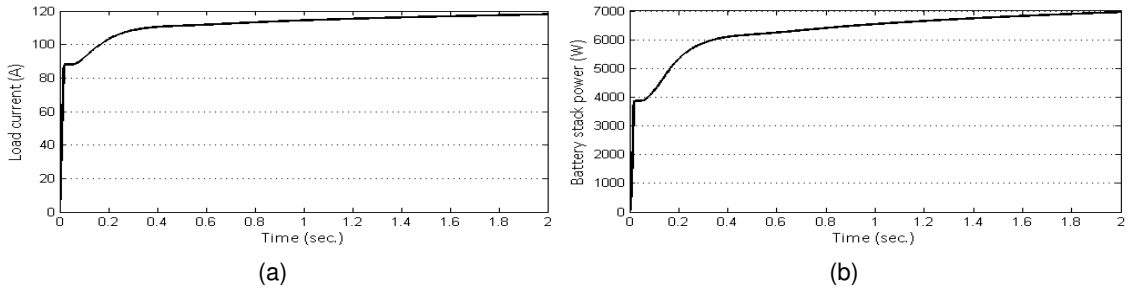


Figure 5.9: SIMULINK simulation results of the connected dc load: delivered electrical (a) current and (b) power.

ures 5.10b and 5.10d, it can be seen that battery is discharged after 100 minutes of charging. It also indicates that after 25 minutes it enters into the over-charging zone. Discharging with a current of $7.2A$ in average, it takes around 35 minutes, which matches with the information available in datasheet, so that the terminal voltage of battery reaches to its cut-off threshold that is around $10.2V$.

Figure 5.10c illustrates the simulation results obtained for wind turbine. In Figure 5.10c, it can be seen that wind turbine generates the maximum lift at $\lambda_{opt} = 8.1$ which is matched with characteristics of the employed wind turbine [94].

The following scenario is carried out to evaluate the stability of the developed model by applying different step changes to manipulated and non-manipulated variables:

- After 100 seconds, load demand sharply changes from 65% to 20% of the maximum value.

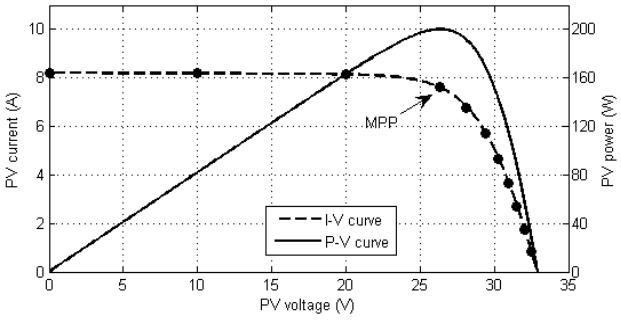
- There is a step change in wind speed at time 350 seconds from 100% to 167% of the rated speed.
- At $t = 300$ seconds, insolation falls to 50% from the initial maximum level.
- The wind turbine pitch angle and the wind branch duty cycles are changed to perform pitch control at $t = 350$ seconds. They are also changed to remove the wind branch shares of generation between $t = 500$ and 600 seconds.

Figure 5.11a illustrates the manipulated variables applied to harvest the maximum power of solar and wind branches, i.e. $2KW$ and $10KW$ respectively. In Figure 5.11a, it can be seen that at $t = 350$ seconds, when wind speed is increased by 67%, pitch angle goes up to 20 degrees and promotes pitching to feather [9]. In effect, although wind speed reaches to 67% above its rated value, the generated power by wind turbine remains stable at the rated value, i.e. $10KW$. Moreover, in Figure 5.11c, it can be seen that despite an initial sharp increase causing by one second delay of the pitch controller delay, rotational speed of wind turbine returns back. Figure 5.11a depicts a sharp decrease of D_w at $t = 500$ seconds that makes the wind turbine share of the electrical power to become zero between $t = 500$ and 600 seconds Figure 5.11b.

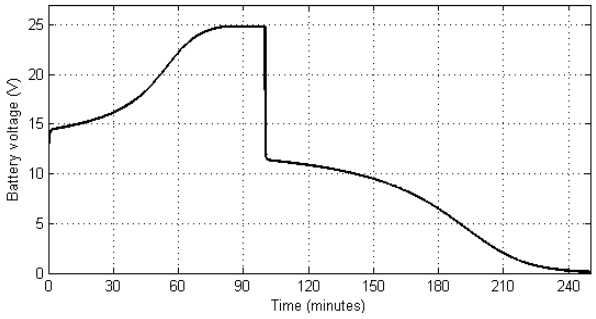
Any discharging to charging mode transition increases the cycle life of battery by $\frac{1}{EFC(DOD_n)}$ as given in Figure 5.11d. From Figure 5.11d, it can be seen that cycle life, as an inherently discontinuous state, is modeled with a continuous approximation. Although battery starts absorbing excess of energy from beginning, its absorbing rate is increased at $t = 100$ seconds after a substantial reduction in load demand. It also compensates energy deficit of the system during the period of time between $t = 500$ and $t = 600$ seconds when wind branch is removed.

Table 5.2: Wind turbine, PMSG, battery stack, and PV parameters in this study.

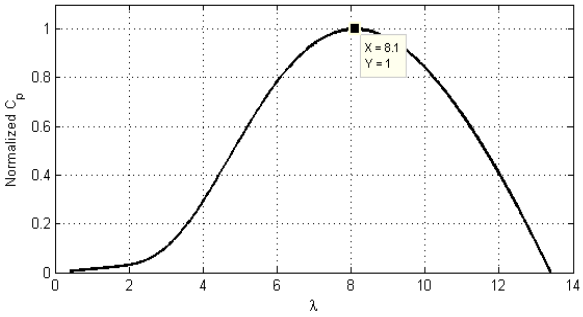
Wind turbine		PMSG		Battery stack		PV array	
$C_1(-)$	0.517	$J(Kg.m^2)$	0.35	$C_{max}(Ah)$	48.15	$R_s(\Omega)$	0.221
$C_2(-)$	116.0	$F(N.m.s)$	0.002	$R_{bat}(\Omega)$	0.019	$R_{sh}(\Omega)$	405.4
$C_3(-)$	0.4	$P(-)$	8	$V_0(V)$	12.3024	$n_d(-)$	1.3
$C_4(-)$	5.0	$\psi(V.s)$	0.8	$P_1(-)$	0.9	$N_s(-)$	54
$C_5(-)$	21.0	$P_{rated}(KW)$	10.0	$N_{bats}(-)$	8	$I_{sc,stc}(A)$	8.21
$C_6(-)$	0.007	$L_s(H)$	0.0083	$N_{batp}(-)$	3	$V_{oc,stc}(V)$	32.9
$\lambda_{opt}(-)$	8.1			$T_s(sec)$	0.726	$k_I(A/K)$	0.003
$P_{wt,nom}(KW)$	10.0			$V_{bstack,nom}(V)$	96.0	$k_V(V/K)$	-0.12
Rad (m)	4.01			$P_{bat,nom}(KW)$	1.296	$N_{pvs}(-)$	1
$U_{x,base}(m/s)$	12.0			$C_{10}(Ah)$	45.0	$N_{pvp}(-)$	10
$C_{p,max}(-)$	0.48			$V_{gas}(V)$	13.0	$P_{pv,nom}(KW)$	2.001



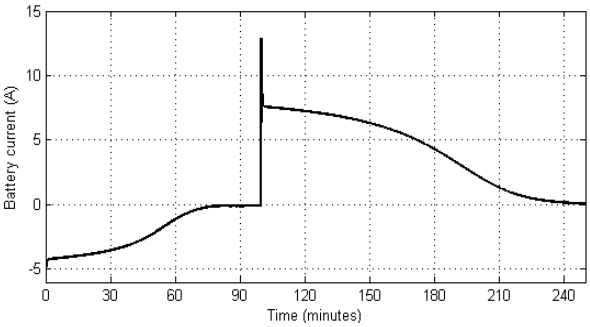
(a)



(b)

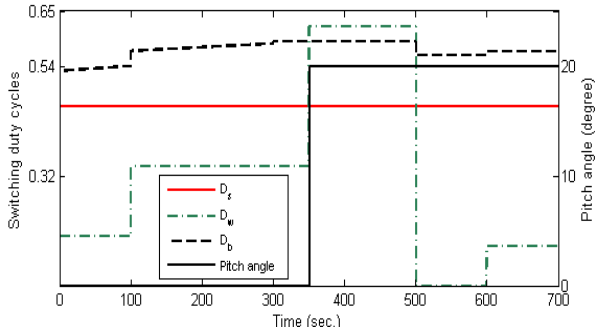


(c)

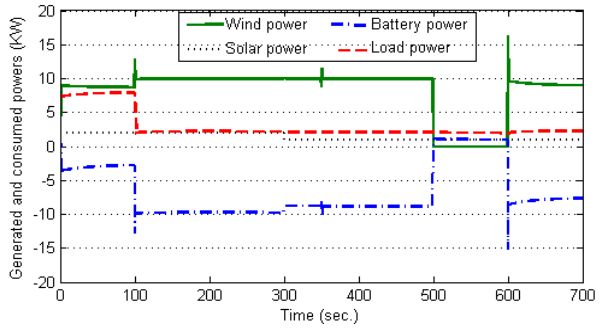


(d)

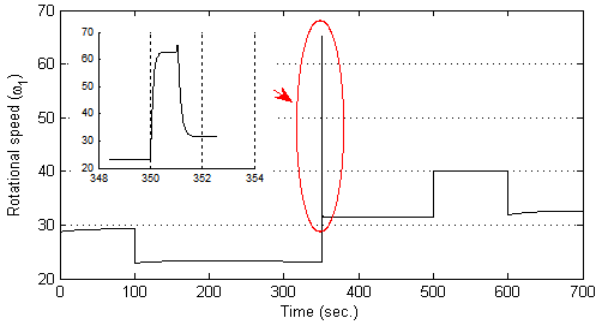
Figure 5.10: Modelica simulation results: (a) $I-V$ (dashed line) and $P-V$ (solid line) curves of KC200GT PV module against experimental points (the circle markers); (b) battery voltage; (c) power coefficient curve of wind turbine; and (d) battery current.



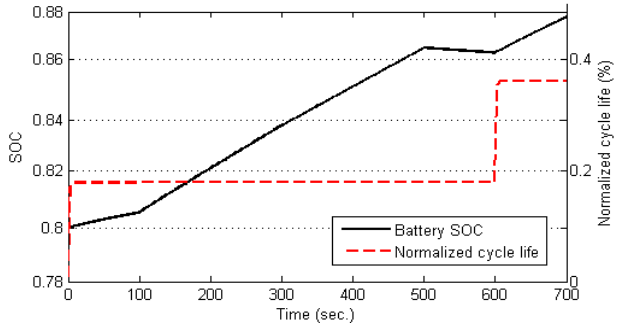
(a)



(b)



(c)



(d)

Figure 5.11: Applied (a) switching duty cycles and pitch angle values; and Modelica simulation results for (b) electrical current of each branch, (c) angular velocity of wind turbine; and (d) cycle life as well as SOC of battery bank.

5.4 Summary

In this chapter, the developed mathematical model in Chapter 2 was used to simulate a sample microgrid. It was shown that it requires employing acausal approach in order to simulate microgrids due to existence of algebraic constraints. A SIMULINK model was developed based on SimPowerSystems toolbox which is suitable to model complex systems in SIMULINK using the acausal approach. It was shown that due to algebraic loops and existence of PWMs, such a simulation is very slow and memory-expensive.

It was shown that the developed Modelica model in Chapter 2 c be solved directly with DASSL general purpose DAE solver which is equipped with event detector and consistent re-initialization features. The obtained results indicated that the proposed model is accurate to simulate different variables of dc microgrids using OpenModelica environment. Moreover, it was shown that the developed model is agile enough to be solved and therefore is suitable for long-term simulation purpose.

Chapter 6

Optimal Energy Management Strategies for Stand-alone dc Microgrids

Coordinated and multivariable energy management strategies are proposed to control the operation of a wind turbine and a photovoltaic array of a stand-alone dc microgrid. As a result, the output power generated by these renewable energy converters are controllable that makes it possible to implement the constant current-constant voltage regime to charge the connected battery bank. The proposed strategies are developed as nonlinear model-predictive controllers that dynamically calculate the optimum pitch angle and three switching duty cycles. It is shown that applying the proposed controller to a sample stand-alone dc microgrid, the battery bank is charged according to the IU charging regime. The variable load demands are also shared accurately between generators in proportion to their ratings. Moreover, the dc bus voltage is maintained at the nominal level with the maximum error of 2.0%.

6.1 Introduction

As explained in Chapter 1, it is more challenging in stand-alone microgrids to address the three well-known voltage regulation, power sharing, and battery management issues which needs to employ more sophisticated control strategies.

The stablity of a dc microgrid can be measured in terms of the stability of its dc

bus voltage level [4, 138]. Therefore, maintaining the dc bus voltage level inside a permissible range, i.e. the voltage regulation, is one of the main objectives of any developed strategy to control dc microgrids [5]. Prior researches (e.g. [4, 138–140]) focused on voltage regulation strategies for the grid-connected microgrids. While the grid voltage source converter (G-VSC) is the primary slack terminal of the grid-connected microgrids [4], storages, such as battery banks, are the only slack terminals that regulate stand-alone or islanded microgrids voltage level. Although battery banks are effective storage systems, their energy absorbing capacity is limited regarding a number of operational constraints, as explained later in this section. The works in [141, 142, 58] present strategies that curtail the renewable power generations if the battery bank cannot absorb the excess generation. However, their curtailment strategy is sensitive to the dc bus voltage variation and is not in coordination with the battery management constraints. Furthermore, their proposed strategies curtail generations of different renewable energy systems (RESs) independently without any proportional power sharing.

When multiple distributed generators (DGs) are connected in parallel, any voltage difference among them leads to a current circulating between generators [143]. In order to prevent the circulating current, load demands need to be shared between all DGs in proportion to their ratings [5, 144]. Moreover, electrical, mechanical, and thermal stresses should be shared evenly among all DGs to prevent overstressing. Droop control is a conventional distributed strategy to share power demands between available generators. It emulates a virtual rotating inertia for an electronically-interfaced generator to mimic the inertia of a conventional synchronous generator [142]. The droop control is extended in [6, 145–148] to be applied to the dc slack terminals by replacing conventional droop curves with a dc power-dc voltage single curve. On the other hand, a dc voltage versus the output current droop curve applicable to the dc microgrids is presented in [5, 143]. Although droop control is very effective, it suffers from at least two drawbacks [1]: i) applying the droop control to the system with non-linear loads causes distortions to the grid voltage. This non-linear load can be a battery bank in the charging mode. ii) It shifts the absolute voltage level of a stand-alone microgrid in terms of the amounts of the load. In order to overcome the second drawback and maintain the microgrid voltage level constant, a hierarchical architecture is introduced in [1, 5, 143], consisting of two extra control

levels; namely, the secondary and tertiary levels.

As mentioned before, there are a number of constraints that limits the battery operation during the charging mode [21]: i) The maximum charging current should be limited in terms of the battery capacity. Applying a constant high level of the charging current increases the battery voltage to reach the gassing voltage fast. ii) Since the internal resistor of the battery increases in terms of the higher state of the charge (SOC), it is essential to reduce the charging current at high SOC values. iii) Although higher charging current reduces the required charging time, battery cannot be fully charged with a constant high level of the charging current. A control strategy is proposed in [138] in which the maximum absorbing power by battery bank is limited as an operational constraint. However, since a battery acts as a non-linear load during the charging mode, limiting the maximum absorbing power does not necessarily limit the charging current. Alternatively, in order to protect battery banks from being overcharged, the work in [141] limits the maximum attainable SOC to be less than 100%. This means that battery banks will never be fully charged. The strategies employed in [6, 149] use an extension of the droop curves to share power demands between different storage units. However, the proposed droop curve does not consider the battery's charging current and gassing voltage constraints. Moreover, applying a linear and static droop curve to a non-linear and dynamic battery bank degrades the power sharing accuracy.

In contrast to the strategies available in literatures where the output power of RESs are not controllable and always equal to the maximum possible power, the proposed strategies curtail, if it is necessary, the power generations of RESs. The EMSs proposed in this work are multivariable NMPC strategies that simultaneously control four variables: i) the wind turbine pitch angle; ii) angular velocity of the wind generator; iii) the PV array operating voltage; and iv) charging current of battery bank. In order to solve the resulting multivariable and non-linear control problem, the proposed EMSs are formulated as novel optimal control problems (OCPs). Unlike strategies which are based on using dump load that only protects battery bank from overcharging, the proposed strategy realizes IU charging regime that helps to increase the battery bank life span. Moreover, removing dump loads, the overall installation cost is reduced.

This chapter is organized as follows: the next section discusses the optimal EMS design issues. It also shows how an EMS is formulated as an OCP. The developed NMPC strategy, that continuously solves the proposed OCP problem, is also presented in Section III. Section IV presents different test scenarios and discusses the obtained results. Finally, the conclusion of the study is given in Section V.

6.2 Design of optimal energy management strategies

In this work, optimal EMSs for stand-alone dc microgrids are formulated as NMPC strategies for comlementarity systems. Such NMPC strategies, as given in in (6.1), are based on a relevant OCP in the form of MPCC for the prediction length of N sample instances. From (6.1), it can be seen that such a relevant OCP includes MCPs, given by (6.1c) besides normal equality and inequality constraints.

$$u^*(.) = \underset{\mathsf{u}(.)\in\Re^n}{\text{arg minimize}} \quad J(\mathsf{x}(n), \mathsf{z}(n), \gamma(n), \mathsf{u}(n); N) :=$$
$$\sum_{k=n}^{n+N} \mathcal{L}(\mathsf{x}(k), \mathsf{z}(k), \gamma(k), \mathsf{u}(k)) + \mathcal{M}(\mathsf{x}(n+N), \mathsf{z}(n+N), \gamma(n+N)), \quad (6.1a)$$

subject to:

$$\mathcal{F}(\mathsf{x}(k), \mathsf{x}(k+1), \mathsf{z}(k), \gamma(k), \mathsf{u}(k), \mathsf{v}(k)) = 0, \quad (6.1b)$$
$$\mathcal{MCP}(\gamma(k), \Gamma(\mathsf{x}(k), \mathsf{z}(k))), \quad (6.1c)$$
$$\mathcal{H}(\mathsf{x}(k), \mathsf{z}(k), \gamma(k), \mathsf{u}(k)) \leq 0, \quad (6.1d)$$
$$\mathsf{x}(n) = \mathsf{x}_0, \mathsf{z}(n) = \mathsf{z}_0, \gamma(n) = \gamma_0, \quad (6.1e)$$
$$\mathcal{R}(\mathsf{x}(n+N), \mathsf{z}(n+N), \gamma(n+N)) = 0, \quad (6.1f)$$
$$\mathsf{x}(k) \in \mathcal{X}, \mathsf{z}(k) \in \mathcal{Z}, \mathsf{u}(k) \in \mathcal{U}, \quad (6.1g)$$
$$0 \leq \gamma(k) \leq 1. \quad (6.1h)$$

Figure 6.1 illustrates a sample of dc microgrids that is controlled with the proposed optimal energy management strategies. For the sake of simplicity, only boost side of the connected bi-directional converter is shown.

From Figure 6.1, it can be seen that the proposed EMSs iteratively get the estimated system states, $\hat{x}$, as the inputs and calculate the optimal solution, $u^*(.)$, as the out-

puts. It is assumed that there is an external state estimator, which is out of the scope of this study, that estimates the current system states, i.e. $\hat{x}$, from the system outputs y. The microgrid dc bus voltage level, V_{dc}, is set externally as a set point $\overline{V}_{dc}$. This means that the developed controller can also acts as a replacement for the secondary and primary levels of the hierarchical architecture introduced in [143].

The non-manipulated variables v, consisting of solar insolation, wind speed, and load demand, need to be predicted N steps ahead to be applied to the developed NMPC. These predicted variables can be obtained from a meteorological center or can be provided by an external module which relies on local measurements and time series techniques like autoregressive-moving-average (ARMA) [150].

The developed NMPC controller consists of three entities: i) the dynamic optimizer that solves the OCP successively; ii) The mathematical model, $\mathcal{F}$, of the system that is used to predict the behavior of the system with respect to the given initial value $\hat{x}$; iii) The cost function and constraints of the relevant OCP which are designed for each operation scenario. The resulting optimal pitch angle, $\bar{\beta}$, is applied as a set point to an inner closed-loop controller. Moreover, the optimal values of the switching duty cycles are applied to the pulse width modulators (PWMs) of dc-dc converters.

The mathematical model of a stand-alone dc microgrid is detailed in Chapter 2 as a set of the following implicit differential and algebraic (DAE) functionals f_i for $i \in \{1, 2, .., 25\}$ and a vector $\mathcal{MCP}$ of two MCPs. For different operation scenarios, there are more functionals to be added to the system model $\mathcal{F}$ in order to model control objectives. These additional functionals, specified for each test scenario, are introduced throughout Section 6.4.

$$\mathcal{F}(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) = \begin{bmatrix} f_1(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \\ f_2(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \\ .. \\ f_{25}(\mathsf{x}, \dot{\mathsf{x}}, \mathsf{z}, \mathsf{u}, \mathsf{v}, \gamma) \end{bmatrix} = 0, \tag{6.2a}$$

$$\mathcal{MCP}(\gamma, \Gamma) = \begin{bmatrix} \mathcal{MCP}_1(\gamma_1, \Gamma_1) \\ \mathcal{MCP}_2(\gamma_2, \Gamma_2) \end{bmatrix}. \tag{6.2b}$$

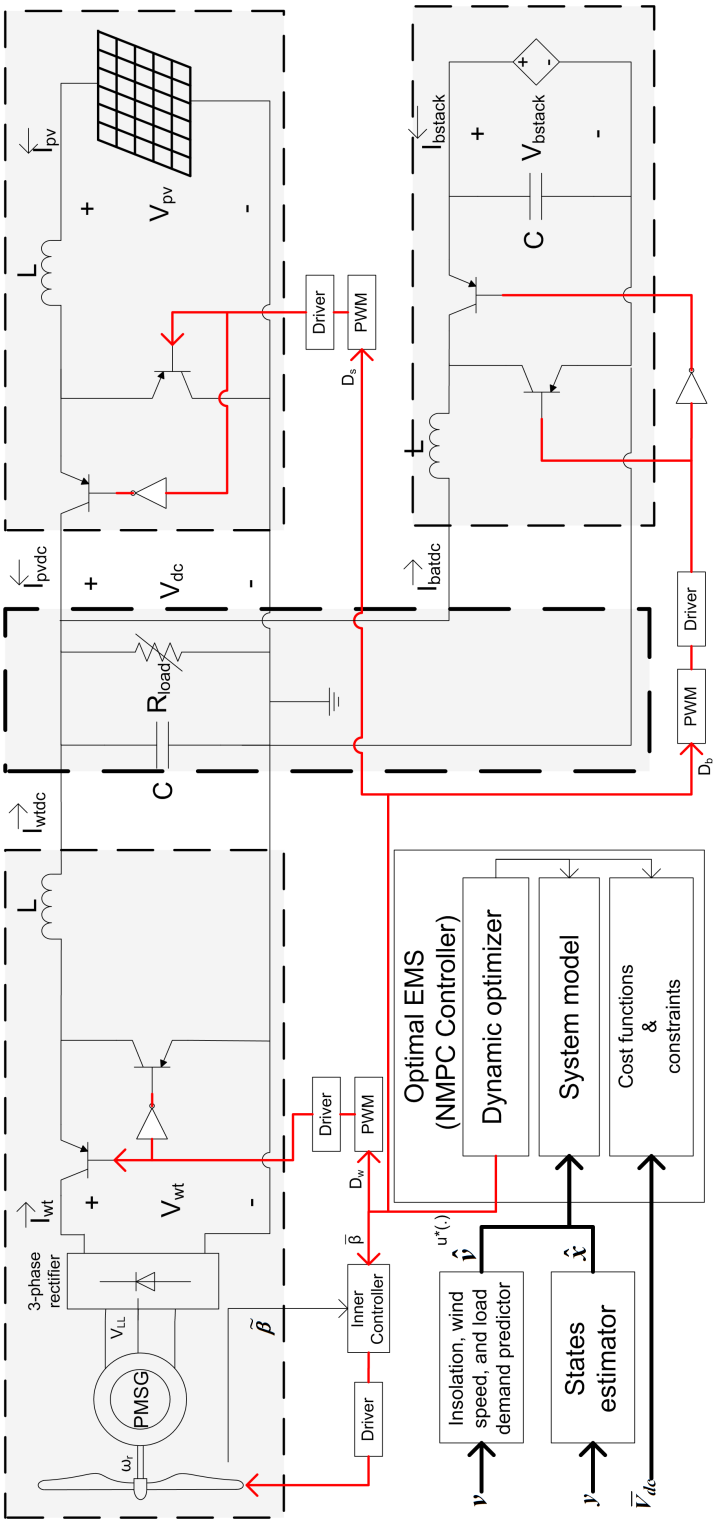


Figure 6.1: A simplified view of dc microgrids and the developed NMPC controller. Battery bank is assumed to work in charging mode.

6.3 Implementation of optimal energy management strategies

The proposed controller employs the collocation method for discretization [25] and is implemented in open-source CasADi environment [81] to benefit from the automatic differentiation (AD) technique and improve its performances [81].

Table 6.1 shows the parameters of different components and their values in this study. These values are used to model the sample dc microgrid depicted in Figure 6.1. The linear load demand is also assumed to be less than or equal to $12\ K$W.

Table 6.1: Components parameters in this study.

Wind turbine		PMSG		Battery stack		PV array	
$C_1(-)$	0.517	$J(Kg.m^2)$	0.35	$C_{max}(Ah)$	48.15	$R_s(\Omega)$	0.221
$C_2(-)$	116.0	$F(N.m.s)$	0.002	$R_{bat}(\Omega)$	0.019	$R_{sh}(\Omega)$	405.4
$C_3(-)$	0.4	$P(-)$	8	$V_0(V)$	12.3024	$n_d(-)$	1.3
$C_4(-)$	5.0	$\psi(V.s)$	0.8	$P_1(-)$	0.9	$N_s(-)$	54
$C_5(-)$	21.0	$P_{rated}(KW)$	10.0	$N_{bats}(-)$	8	$I_{sc,stc}(A)$	8.21
$C_6(-)$	0.007	$L_s(H)$	0.0083	$N_{batp}(-)$	3	$V_{oc,stc}(V)$	32.9
$\lambda_{opt}(-)$	8.1			$T_s(sec)$	0.726	$k_I(A/K)$	0.003
$P_{wt,nom}(KW)$	10.0			$V_{bstack,nom}(V)$	96.0	$k_V(V/K)$	-0.12
Rad (m)	4.01			$P_{bat,nom}(KW)$	1.296	$N_{pvs}(-)$	1
$U_{x,base}(m/s)$	12.0			$C_{10}(Ah)$	45.0	$N_{pvp}(-)$	10
$C_{p,max}(-)$	0.48			$V_{gas}(V)$	13.0	$P_{pv,nom}(KW)$	2.001

Table 6.2 summarizes the developed NMPC controller in this study that optimally manages the energy flow across stand-alone dc microgrids. It illustrates that the different equations described throughout Chapter 2 construct the system model $\mathcal{F}$ and makes it possible for the controller to predict the near future behavior of dc microgrid. The non-manipulated variables, i.e. wind speed, insolation, and load demand, are modeled as parameters which are initialized for every prediction horizon. The ambient temperature T_x is assumed to be constantly equal to $298.15\ K$, for simplicity's sake.

6.4 Test scenarios

In order to evaluate the performance of the developed optimal EMS in terms of different control objectives, three different test scenarios are carried out throughout this section. The first test scenario shows a multivariable MPPT strategy for stand-alone dc microgrids as hybrid systems. It indicates general operations of a load tracking strategy for such a hybrid system. In order to solve the resulting OCP with a general purpose NLP solver, it is assumed that battery bank only switches between

Table 6.2: A pseudocode of the proposed optimal energy management strategy.

Step1: Configurations

Set $\overline{\alpha}_1 = 0.02\overline{V}_{dc}$ and $\overline{\alpha}_2 = 0.01$; (the values of the slack variables in Eqs. (6.7)-(6.8))

Set $T = 10.0(sec)$, $N = 2$; (prediction horizon and the number of samples, respectively)

Instantiate an NLP solver; (non-linear optimization problem solver)

Step 2: Measurements

Measure the differential states I_f, Q_{bat}, and ω_r values form the system;

Measure the dc bus voltage V_{dc};

Measure the battery stack voltage level V_{bstack};

Get the predicted wind speed, insolation, and load demand for the next 10 seconds form an external estimator;

Step 3: Constructing the OCP given by Eq. (6.1) in CasADi environment [81]

IF $V_{bstack} < V_{gas}N_{bats}$ THEN

Use Eq. (6.9a) as the cost function; (constant current charging regime)

ELSE use Eq. (6.9b) as the cost function; (constant voltage charging regime)

Construct the vector $\mathcal{F}$ (Eq. 6.1b) using functionals $f_1 - f_{27}$ (as in Eqs. (2.2)-(2.14) and (6.7)-(6.8)) ;

Apply the current and next step values of the wind speed, insolation, and load demand as the OCP parameters;

Construct Eq. (6.1g) from the box constraints defined by Eqs. (6.10)-(6.11) as well as two constraints $|\alpha_1| \leq \overline{\alpha}_1$ and $|\alpha_2| \leq \overline{\alpha}_2$;

Step 4: Initializations

Set the measured values of I_f, Q_{bat}, and ω_r as the initial values of differential states

Set the measured value of V_{dc} as the initial value of dc bus voltage level

Step 5: Discretization and solving the discretized problem

Discretize the OCP problem using the collocation method

Construct equivalent NLP problem

Solve the equivalent NLP problem using standard NLP solvers to calculate the optimal solution $u^*(.)$

Step 6: Applying the control variables

Constructing the control law using the first sample of the optimal solution, i.e. $u^*(0)$;

Apply the control variable according to the above control law ; (the first sample of the optimal solution is applied to the system)

GOTO Step 2: Measurements ;

the charging and discharging modes of operation at the edge of the discretized time slots.

The next two test scenarios depict the controller effectiveness to implement a constant current and constant voltage charging regime. A practical EMS can switch between these two scenarios depending on the battery bank voltage level.

6.4.1 Scenario I: Multivariable load tracking strategy for stand-alone dc microgrids

Controller design

In order to use ordinary NLP solvers, the MCP constraint of the system model, given in (6.2b), can be regularized as the following equality and inequality constraints [25]:

$$\begin{aligned} \Gamma(\gamma) - w + v = 0, \\ w, v \geq 0, \quad w^T\gamma = \epsilon, \quad v^T(1-\gamma) = \epsilon. \end{aligned} \tag{6.3}$$

where $0 < \epsilon \ll 1$ is a design parameter and $\Gamma(\gamma) = [\Gamma_1(\gamma_1) \quad \Gamma_2(\gamma_2)]^T = [I_{bstack} \quad -t_{chg}]^T$ and $\gamma = [\gamma_1 \quad \gamma_2]^T$ are respectively the functions and slack variables given in (2.5) and (2.7). Moreover, $w = [w_1 \quad w_2]^T$ and $v = [v_1 \quad v_2]^T$ are slack variables, defined to regularize MCPs.

Control objectives

The developed energy management strategy in this chapter addresses two control objectives: i) maximizing the battery bank SOC; and ii) regulating dc bus voltage level of the microgrid. These objectives are formulated as the cost function J and a path constraint h_1, respectively.

The cost function J, given in (6.4), is the battery bank SOC deviation from the fully charged state which should be minimized in terms of manipulated control signals u and slack variables w, v, and γ.

$$J(\mathsf{x}(n), \mathsf{z}(n), \mathsf{u}(n), w(n), v(n), \gamma(n); N) := \sum_{k=n}^{n+N} \|SOC(k) - 1.0\|_2 + \|SOC(n+N) - 1.0\|_2 . \tag{6.4}$$

The permissible deviation of the dc bus voltage level V_{dc} from the specified set point $\overline{V}_{dc}$ is given in (6.7) as a slack variable α_1 which is a design parameter.

$$h_1 : \quad |V_{dc} - \overline{V}_{dc}| - \alpha_1 \leq 0. \tag{6.5}$$

In this test scenario, the design parameter α_1 is set to be in the range of $\pm 2\%$ of the $\overline{V}_{dc}$. It means that for a set point of 48.0 volt, the maximum permissible deviation is ± 0.96 volt.

Control constraints

The generated power by wind turbine should be limited to its rated power by employing a pitch angle controller [9]. This requirement is addressed by the following path constraint:

$$h_2 : \quad -T_e \omega_r - P_{wt,nom} \leq 0. \tag{6.6}$$

The variation of the wind turbine angular velocity, i.e. $\Delta(\omega_r)$, is also restricted in order to decrease the tower vibrations [9]. Moreover, the charging and discharging currents of each battery are limited respectively to $0.16 C_{10}$ and $0.23 C_{10}$ amperes, where C_{10} is the nominal capacity of each battery given in Table 6.1.

Results and discussion

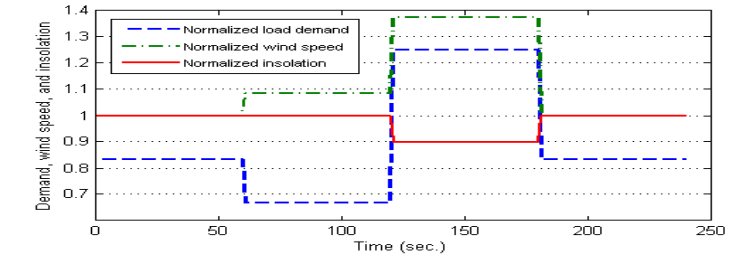
Non-manipulated variables, i.e. wind speed, insolation, and load demand, are applied to the system according to the given profiles in Figure 6.2a. Figure 6.2b, on the other hand, illustrates the calculated optimal pitch angle and one of the switching duty cycles, i.e. D_w. In response to substantial wind speed rises during the time between 60 and 180 seconds, the wind turbine pitch angle is increased to around 7 and then 14 degrees. Figure 6.2c shows that despite an increase of 38% by wind

speed, the generated power remains steady at the rated power as the result of the developed coordinated pitch and variable speed control strategy. From Figure 6.2d, it can be seen that in order to maintain such a constant output power, the optimal switching duty cycle and pitch angle are calculated to decline the normalized power coefficient to 0.393.

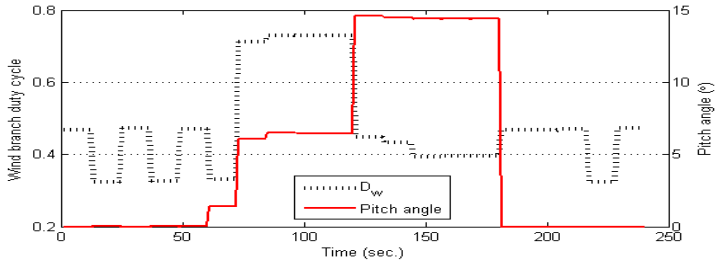
Figure 6.2c shows that the developed strategy acts as MPPT algorithm during usual operation. However, there is an exception during the period of time between 60 to 120 seconds, when the load demand falls down. Since the maximum permissible charging current of batteries is specified to be $0.16C_{10}$, the controller curtails the amount of the generated power to meet the charging constraint. Figure 6.2e depicts that the charging current of each battery remains steady at $0.16C_{10}$, i.e. $7.2\ A$.

Moreover, from Figure 6.2a, it can be seen that the load demands fluctuate between 68% and 125% of the rated value causing battery bank switches between the charging and discharging operation modes. The consequent variations of the slack variables γ_1, w, and v are given in Figure 6.3a. From Figure 6.3a, it can be seen that battery bank switches from charging mode ($\gamma_1 = 1$) to discharging mode ($\gamma_1 = 0$) at $t = 120$ seconds in order to maintain the dc bus voltage level in the permissible range. Figure 6.3b illustrates that despite a substantial increase of load demand at $t = 120$ seconds, dc bus voltage level remains in the range of $48.0 \pm 2\%$. Moreover, the given variables w and v in (6.3), respectively, to model the instantaneous changes of the battery current direction, switch between 0 and the charging and discharging currents of batteries. Since the charging and discharging currents of batteries are respectively limited to $0.16C_{10}$ and $0.23C_{10}$ amperes, the slack variables w and v are also bounded by these values.

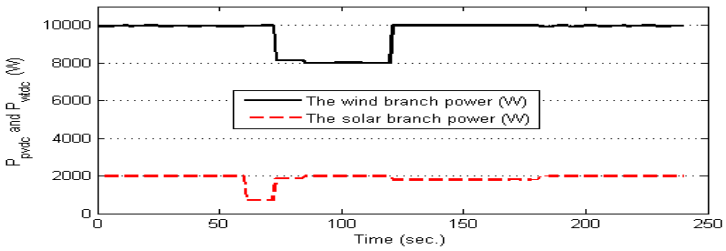
Finally, Figure 6.3c illustrates that switching between two modes of the battery bank operation changes SOC trends during the simulation time. The SOC value starts going down at $t = 120$ seconds, as a result of the battery bank discharging. Although the developed control strategy tries to harvest the maximum available wind and solar energies and maximize the SOC of battery bank, it may also curtail generations in some cases. Due to this generation curtailment, SOC rises identically between $t = 0$ and 120 seconds in spite of wind speed and load demand variations.



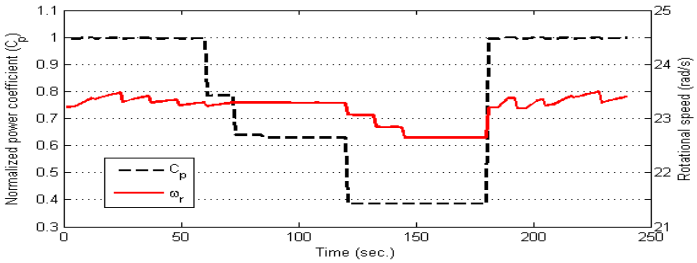
(a)



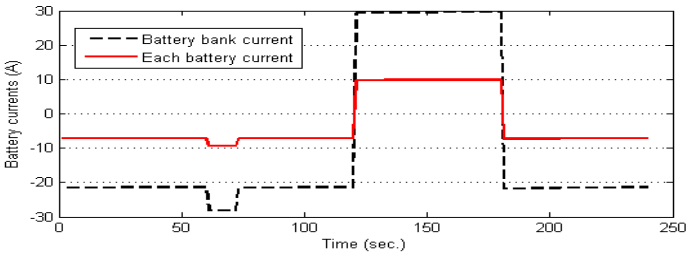
(b)



(c)

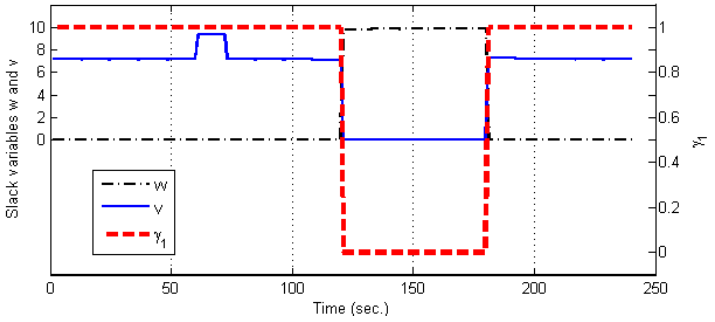


(d)

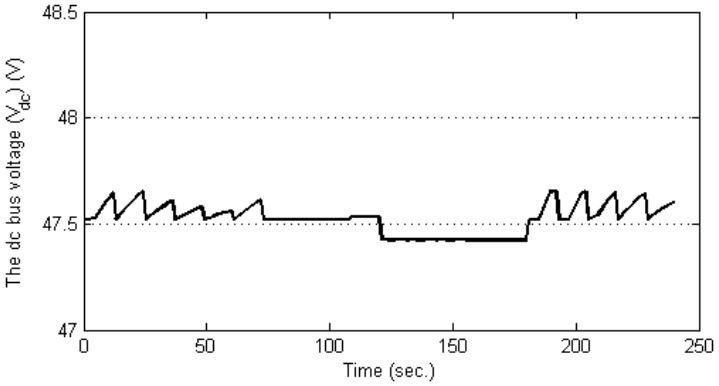


(e)

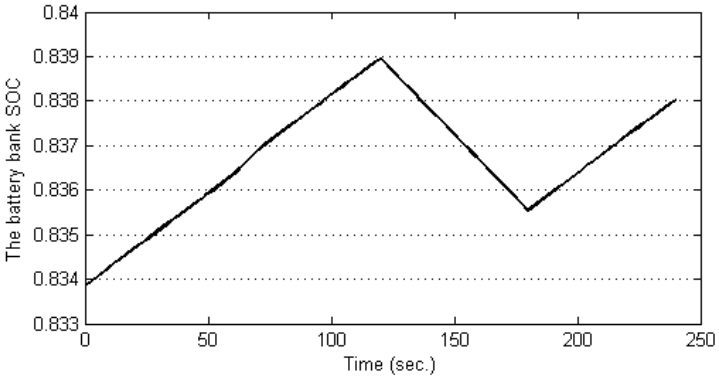
Figure 6.2: (a) Normalized values of the applied non-manipulated variables; (b) the optimal switching duty cycle and pitch angle of wind branch; and the resulting (c) powers by wind and solar branches, (d) batteries current, and (e) angular velocity and normalized power coefficient of wind turbine.



(a)



(b)



(c)

Figure 6.3: (a) Switchings events of slack variables and the resulting (b) dc bus voltage level of microgrid and (c) SOC trends.

6.4.2 Scenario II: Constant current charging mode

This scenario covers the following three different cases which are run successively:

- Case I: wind turbine and PV array generate enough power at their MPPs to supply load demands and charge battery bank with its nominal charging current.
- Case II: the generated power is just enough to supply the load demands and therefore battery bank is not charged or is charged with the current less than its nominal charging current.
- Case III: the generated power is more than the required power to supply the load demands and charge battery bank with its nominal charging current.

Each case lasts for 5 minutes and therefore the total period of the simulation time is 15 minutes. In order to calculate the optimal control variables every 5 seconds, the developed NMPC controller runs exactly 60 times as per each case.

Control objectives

The developed EMS for this test scenario addresses three control objectives: i) regulating the dc bus voltage level of microgrid; ii) power sharing between wind turbine and PV array in proportion to their ratings; and iii) managing the battery to be charged according to the IU charging regime. These objectives are formulated as a combination of two slack variables and the cost function J.

The permissible deviation of the dc bus voltage level V_{dc} from the specified set point $\overline{V}_{dc}$ is given in (6.7) in the form of a slack variable α_1 which is a design parameter. In this study, α_1 is set to be in the range of $\pm 2\%$ of $\overline{V}_{dc}$. It means that for a set point of 48.0 volt, the maximum permissible deviation is ± 0.96 volt.

$$f_{26} = \alpha_1 - (V_{dc} - \overline{V}_{dc}). \tag{6.7}$$

The second part of the above mentioned ternary objective, i.e. the power sharing criterion, is also formulated as a constraint in (6.8). The design parameter α_2 indicates the permissible deviation from the proportional power sharing criteria. If α_2 is

zero, the generated power ratio between the wind and PV branches is equal to the ratio between their nominal powers, $P_{wt,nom}$ and $P_{pv,nom}$ which is 5. The generated powers are also normalized with respect to the wind speed and insolation values, i.e. U_x and S_x. In this study, the design parameter α_2 is set to be in the range of $\pm 1\%$ that increases the flexibility of the algorithm with the cost of slight penalty.

$$f_{27} = \alpha_2 - \left\{ \frac{I_{wtdc} V_{dc}}{P_{wt,nom}} \left(\frac{U_{x,base}}{max(U_x, U_{x,base})} \right)^3 - \frac{I_{pvdc} V_{dc}}{P_{pv,nom}} \frac{S_{x,base}}{max(S_x, S_{x,base})} \right\}. \tag{6.8}$$

The last part of the above mentioned ternary objective, i.e. battery bank charging management, is considered as the cost function. There are two separate cases and the presented EMS switches between them depending on the battery voltage level: i) the battery voltage level is less than its gassing voltage; and ii) the battery voltage level exceeds the gassing voltage. (6.9a) and (6.9b) are the discretized cost functions designed for these two separate cases where $\|.\|_2$ denotes Euclidean distance operator. The cost functions are convex combinations of two objectives with the weights β_1-β_4. While β_1 and β_3 are close to 1.0, β_2 and β_4 are close to zero. In order to keep dc bus voltage level as close as possible to $\overline{V}_{dc}$, its deviation is also added to the cost function. Otherwise, dc bus voltage level may stay at the upper or lower boundaries of the permissible range of $V_{dc} \pm 0.96$ volt.

$$\begin{aligned} J\big(\mathsf{x}(n), \mathsf{z}(n), \mathsf{u}(n), N\big) := \\ \sum_{k=n}^{n+N} \left\{ \beta_1 \left\| \frac{1}{\overline{I}_c} \Big(\frac{I_{bstack}(k)}{N_{batp}} - \overline{I}_c \Big) \right\|_2 + \beta_2 \left\| \frac{1}{\overline{V}_{dc}} \Big(V_{dc}(k) - \overline{V}_{dc} \Big) \right\|_2 \right\} + \\ \left\{ \beta_1 \left\| \frac{1}{\overline{I}_c} \Big(\frac{I_{bstack}(N)}{N_{batp}} - \overline{I}_c \Big) \right\|_2 + \beta_2 \left\| \frac{1}{\overline{V}_{dc}} (V_{dc}(N) - \overline{V}_{dc}) \right\|_2 \right\}, \end{aligned} \tag{6.9a}$$

$$\begin{aligned} J\big(\mathsf{x}(n), \mathsf{z}(n), \mathsf{u}(n), N\big) := \\ \sum_{k=n}^{n+N} \left\{ \beta_3 \left\| \frac{1}{N_{bats} V_{gas}} \Big(V_{bstack}(k) - N_{bats} V_{gas} \Big) \right\|_2 + \beta_4 \left\| \frac{1}{\overline{V}_{dc}} \Big(V_{dc}(k) - \overline{V}_{dc} \Big) \right\|_2 \right\} + \\ \left\{ \beta_3 \left\| \frac{1}{N_{bats} V_{gas}} \Big(V_{bstack}(N) - N_{bats} V_{gas} \Big) \right\|_2 + \beta_4 \left\| \frac{1}{\overline{V}_{dc}} \Big(V_{dc}(N) - \overline{V}_{dc} \Big) \right\|_2 \right\}. \end{aligned} \tag{6.9b}$$

Battery bank is charged with the constant current $\overline{I}_c$ when its voltage is less than gassing voltage. Moreover, the voltage level of battery bank should always remain below its gassing voltage $\overline{V}_{gas}$ to protect it from permanent damages or even probable explosion.

Box constraints

The generated power by wind turbine must be limited to its rated value specified by manufacturers. In order to regulate the produced aerodynamic power and comply with limitations, it requires employing a pitch angle controller [9]. This requirement is addressed by introducing the given box constraint in (6.10) on the mechanical power generated by wind turbine.

$$0 \leq -T_e \omega_r \leq P_{wt,nom}. \tag{6.10}$$

Other box constraints on the manipulated variables as well as system states are formulated as follows:

$$\mathsf{x}_{min} \leq \mathsf{x} \leq \mathsf{x}_{max}, \tag{6.11a}$$

$$\mathsf{u}_{min} \leq \mathsf{u} \leq \mathsf{u}_{max}, \tag{6.11b}$$

where x_{min} and u_{min} specify lower boundaries for system states and manipulated variables, respectively. x_{max} and u_{max} are also the upper boundaries of these variables. For instance, the switching duty cycles are limited between 20 and 80% and the pitch angle should be less than 30 degrees.

Initial constraints

Prior to calculating the optimal solution of the next receding horizon N, all system states, including angular velocity of wind turbine, actual available charge of battery Q_{act}, and the filtered battery current, I_f, are initialized by the measured or estimated values. Moreover, dc bus voltage level is measured and applied to controller as an

initial value for V_{dc}. These measured or estimated values construct initial constraints of the OCP, as given in (6.1e).

Results and conclusion

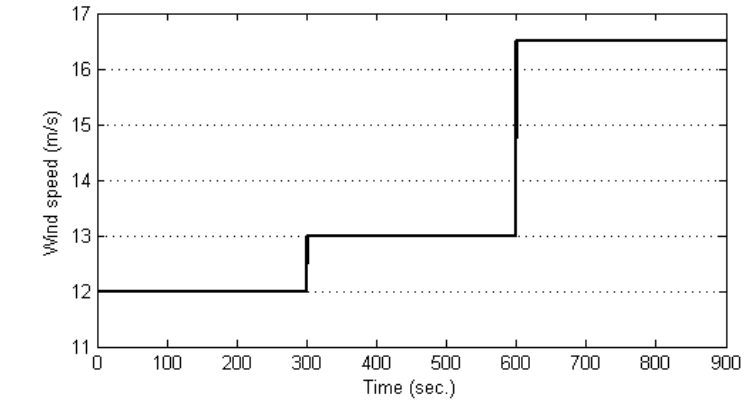
Table 6.3 summarizes design parameters of the developed NMPC controller. The number of discretization samples N is chosen to be 2 and therefore one only needs to predict the next sample of non-manipulated variables v. The reported computational times of a 10-second prediction horizon is calculated on an Intel CORE 2 DOU machine with $3GB$ of RAM. These computational times indicate that choosing a 5-second sampling time h provides adequate margin for all calculations of the dynamic optimizer module in Figure 6.1. Moreover, it means that the microgrid voltage level deviation from set points is evaluated every 5 seconds that complies with the hierarchical architecture specifications [143].

Figure 6.4 illustrates non-manipulated inputs to the system including wind speed, insolation, and load demand. From Figure 6.4, it can be seen that during the first 5 minutes, i.e. case I, wind speed as well as insolation are equal to the rated values and load demand is less than the maximum available power. On the other hand, during the second 5 minutes wind speed exceeds its rated value while load demand reaches to the maximum rated power, i.e. $12\ K$W. Finally, while wind speed sharply exceeds its rated value during the last case, load demand is also decreased below the maximum available power. Without loss of generality, wind speed and load demand are assumed to be constant within each case, i.e. for 5 minutes. Moreover, solar irradiance is constant during the simulation only for results clarification.

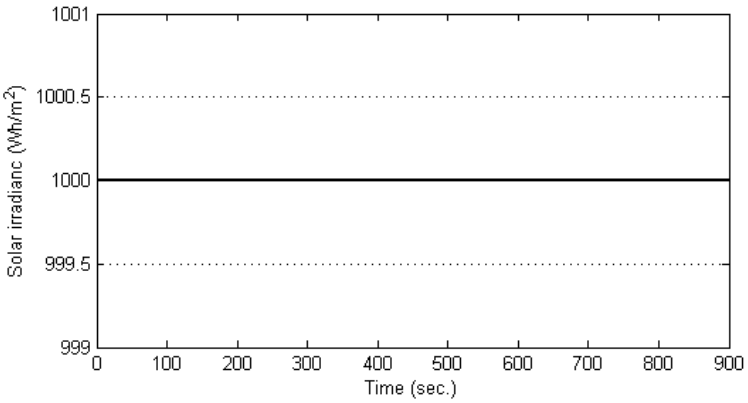
The optimal control variables, $u^*(.)$, minimize the cost function J given by (6.9). Figure 6.5 depicts these calculated optimal control variables for the non-manipulated inputs in Figure 6.4. Applying these optimal control variables to stand-alone dc microgrid, the resulting variables of wind and solar branches are depicted in Figure 6.6.

Table 6.3: Design parameters and computational times for the developed NMPC controller.

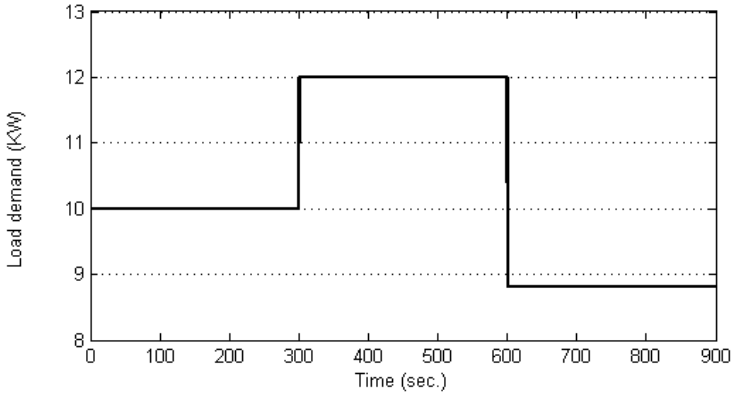
Parameter Name	Parameter Value	Parameter Name	Parameter Value
Prediction horizon T (sec)	10	Average Computational Time (sec)	2.066
Sampling time h (sec)	5.0	Minimum Computational Time (sec)	0.628
No. of the discretization samples N	2	Maximum Computational Time (sec)	3.565
$\overline{V}_{dc}(V)$	48.0		



(a)

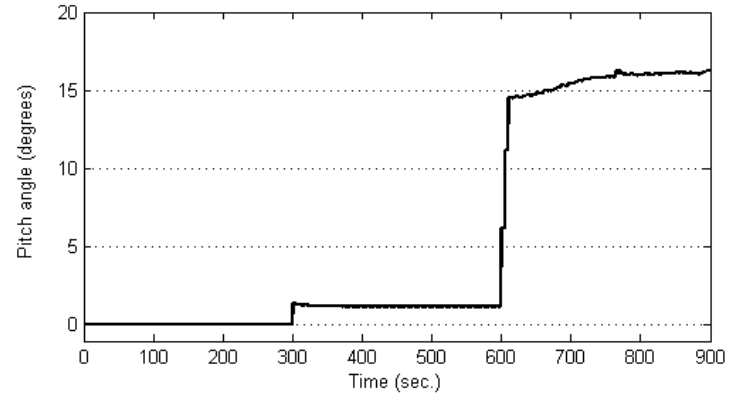


(b)

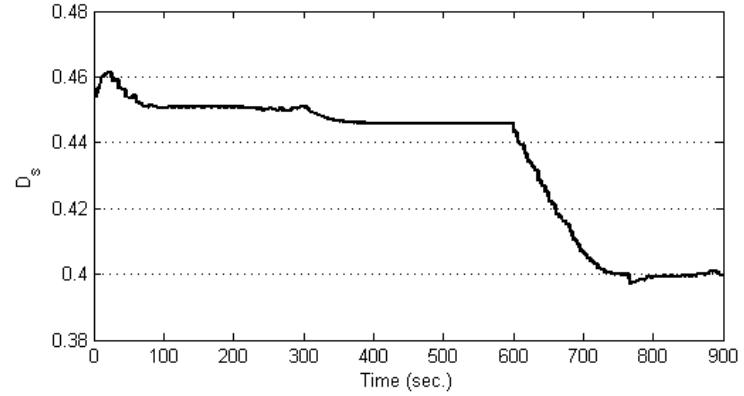


(c)

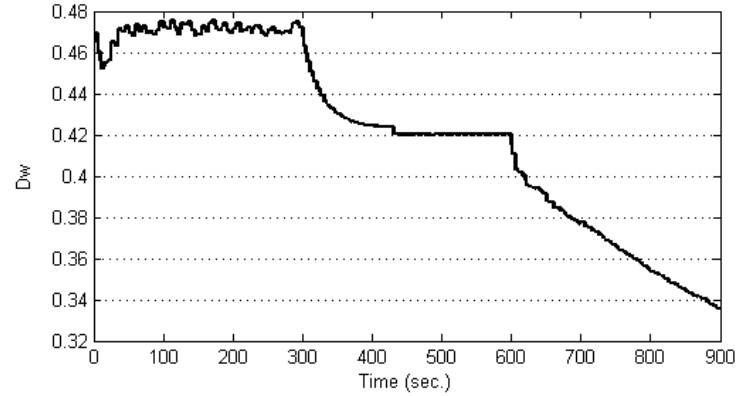
Figure 6.4: Non-manipulated input variables: (a) wind speed, (b) solar irradiance, and (c) load demand.



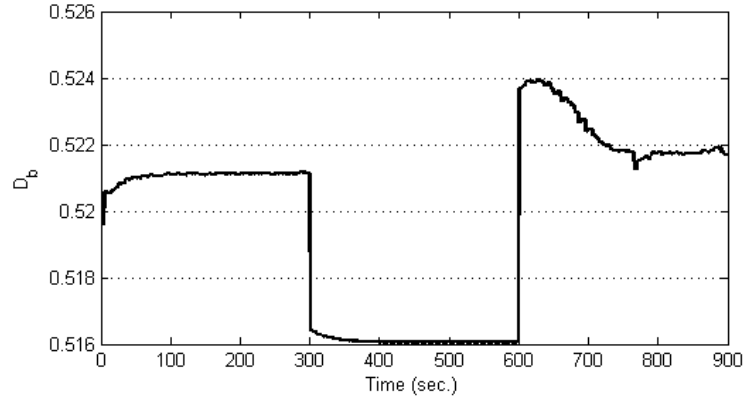
(a)



(b)



(c)



(d)

Figure 6.5: Calculated optimal control variables: (a) pitch angle, switching duty cycles of (b) solar-, (c) wind-, and (d) battery-branch converters.

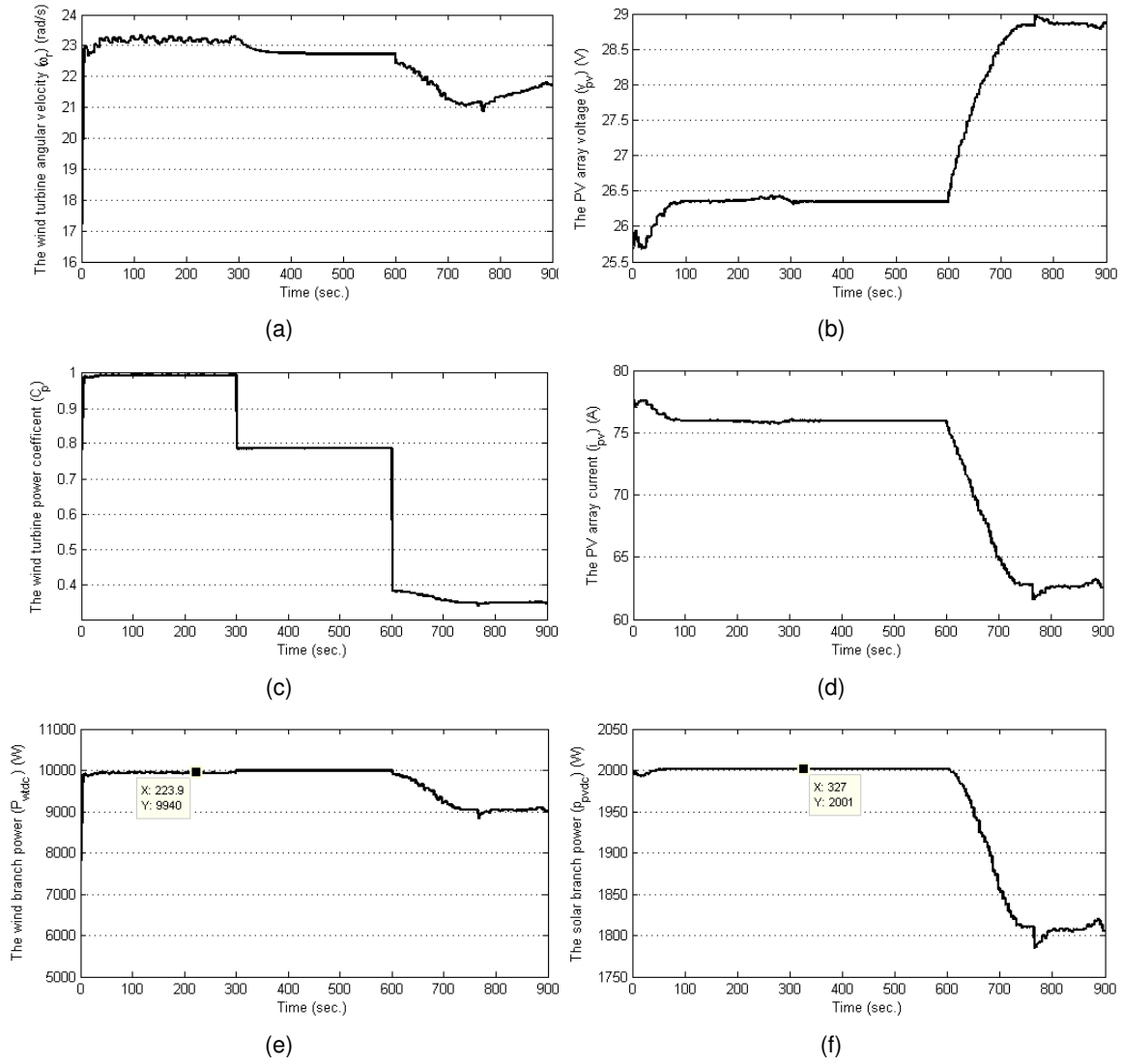


Figure 6.6: Different variables of wind and solar branches: (a) wind turbine angular velocity, (b) PV array voltage, (c) wind turbine power coefficient, (d) PV array current, and (e)-(f) generated power by each branch.

Moreover, Figure 6.10 illustrates dc bus voltage level, supply and demand currents, as well as SOC and charging current of battery bank.

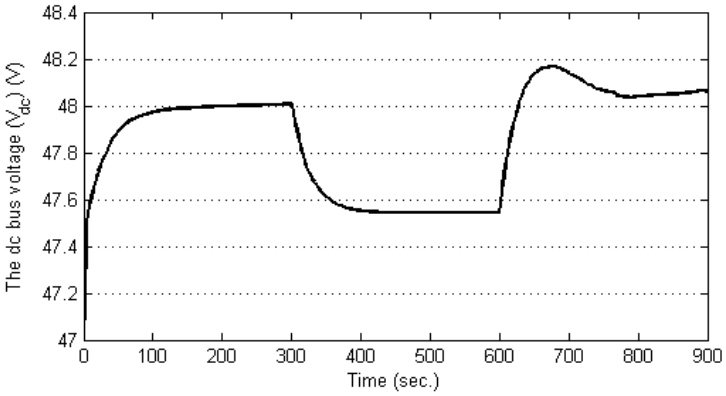
Wind branch operates at its MPP during case I, i.e. before $t = 300$ seconds. Since wind speed is equal to its rated value (Figure 6.4a), pitch angle is calculated to be zero as given by Figure 6.9a. Figure 6.9c shows that in order to harvest the maximum available power (Fig. 6.6e), controller adjusts the switching duty cycle of the connected buck converter, D_w. The resulting rotational speed of wind turbine is close to its nominal value as given by Figure 6.6a that produces the maximum power coefficient (Figure 6.6c). From Figures 6.4a and 6.5a, it can be seen that at $t = 300$ and 600 seconds, when wind speed exceeds its rated value, pitch angle goes up to 1.2 and 16 degrees, respectively, that promotes pitching to feather [9]. It ensures

that wind turbine works at its rated power during case II. However, in order to curtail its generation during case III, i.e. during the period of time after $t = 600$ seconds, controller not only promotes pitching to feather but also reduces rotational speed. Figures 6.6c and 6.6e, respectively, indicate that as a result, power coefficient and generated power of wind turbine are declined, respectively, to 0.348 and 9.039 KW.

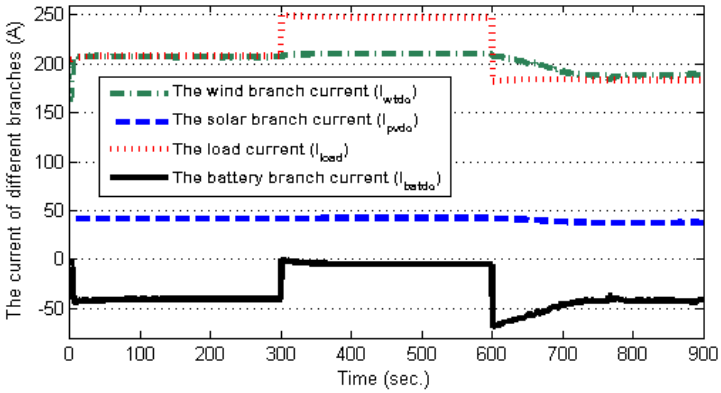
Solar branch also works at its MPP during cases I and II (Figure 6.6f); however, its generated power is curtailed during case III. From Figures 6.6b and 6.6d, it can be seen that though PV array operates at its MPP, i.e. $V_{pv} = 26.3V$ and $I_{pv} = 76.1A$, during cases I and II, controller changes its operation point during case III in order to curtail its generated power down to 1.808 KW. Dividing this value to the curtailed wind branch power, i.e. 9.039 KW, it reveals that controller shares the curtailed power between two branches in proportion to their rating. The power sharing accuracy during case III is 0.035% which is within the permissible range of $\pm 1\%$, as defined by $\overline{\alpha}_2$ in Table 6.2. It should be noted that though both wind and solar branches work at their maximum power points during case I, the generated power by the wind branch is slightly less than the nominal power at its rated speed. This slight difference can be reduced by decreasing the design parameter $\overline{\alpha}_2$.

In spite of a significant variation of wind speed and load demand, Figure 6.7a indicates that dc bus voltage of microgrid is maintained within its permissible range, which is $\overline{V}_{dc} \pm 0.02\overline{V}_{dc}$. From Figure 6.7a, it can be seen that during case II, when there is not enough generated power to charge battery, controller reduces dc bus voltage level to reduce the wasted energy across lines. However, beginning case III, voltage level returns back to its nominal value of $48.0V$. Figure 6.7b shows the supplied and demanded currents during the period of the simulation time.

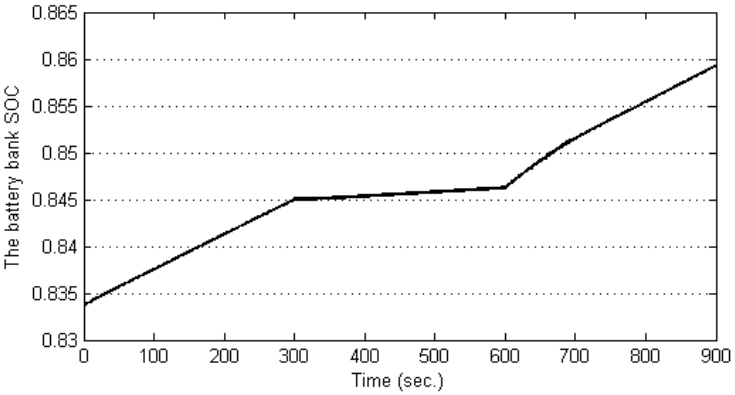
In order to fully charge battery bank, the developed EMS does not let the charging current to exceed its nominal value. The nominal charging current is assumed to be $0.15C_{10}$ which is 6.75 A per each battery and 20.25 A for the battery bank. Figure 6.7d depicts that charging current of battery bank remains constant at its nominal value during cases I and III. The optimal EMS curtails the generated power such that charging current, which initially exceeds its nominal value during case III, is fixed later. Although there is not enough power during case II to charge battery at its nominal current, charging current is still constant. In Figure 6.7c, it can be seen



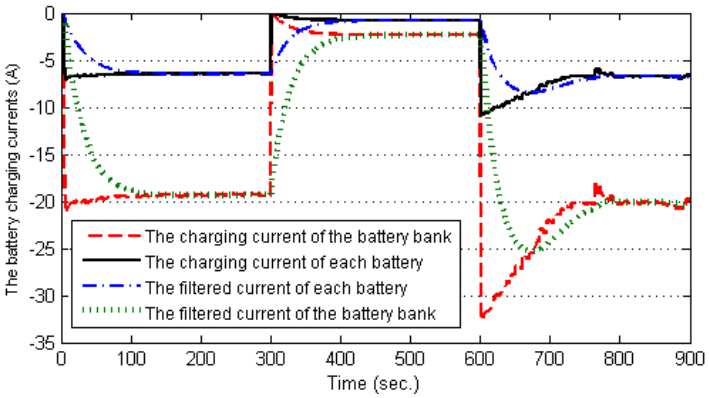
(a)



(b)



(c)



(d)

Figure 6.7: (a) Dc bus voltage of microgrid, (b) supply and demand currents of different components, (c) SOC, and (d) charging current of battery bank.

that this strategy helps battery to be charged up to high SOC values.

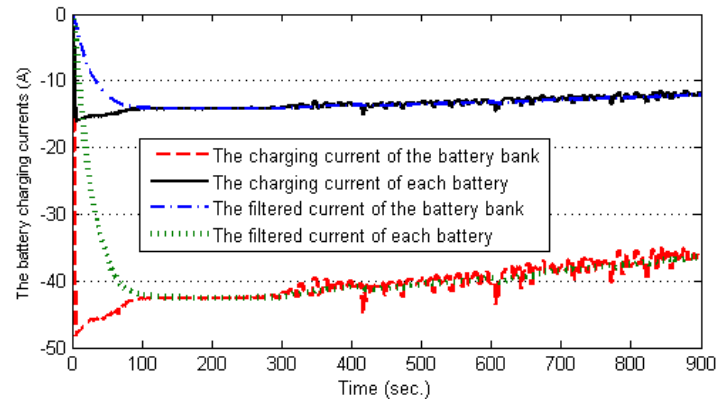
6.4.3 Scenario III: Constant voltage charging mode

Terminal voltage of battery bank rises by Scenario II due to constant charging currents. Once the battery terminal voltage level reaches its gassing voltage, charging current should be gradually reduced in order to prevent exceeding gassing voltage threshold. This constant voltage charging strategy helps battery bank to be fully charged without the risk of permanent damage. From Table 6.2, it can be seen that the developed controller switches to the constant voltage charging mode when terminal voltage of battery bank reaches to its gassing voltage. For this purpose, the cost function of the developed NMPC strategy is switched as given in Step 3 of Table 6.2.

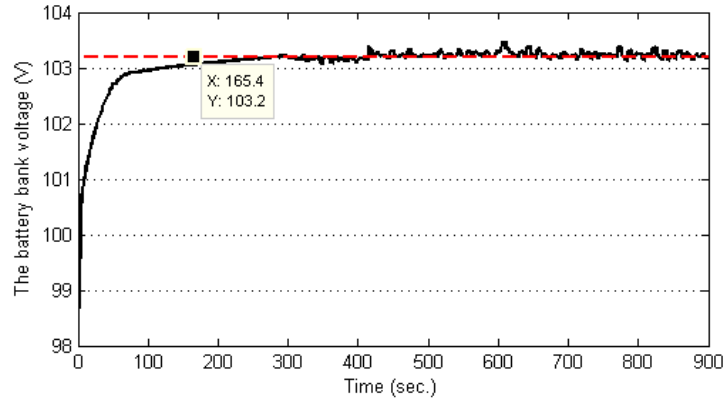
Results and conclusion

Scenario III operates at wind speed and insolation levels similar to Scenario II, which are given by Figure 6.4. Figures 6.8a and 6.8b, respectively, show the charging current and terminal voltage variations of battery bank. From Figure 6.8a, it can be seen that battery bank is charged with a constant current equals to $0.31C_{10}$ during the period of time up to $t = 300$ seconds when its terminal voltage reaches to the gassing voltage, i.e. $41.85\ A$. The developed controller does not let charging current to exceed this nominal value. By reaching the gassing voltage at $t = 300$ seconds, controller starts gradually reducing charging current in order to maintain the battery bank voltage constant. As a safe margin, terminal voltage of each battery is maintained around 0.8%, or equivalently $0.1\ V$, less than its gassing voltage. In other words, the terminal voltage of battery bank stays constant at 103.2 V for the gassing voltage of 13.0 V as per each battery unit. Figure 6.8c indicates that battery can be fully charged with the constant current-constant voltage charging strategy with no risk of exceeding its gassing voltage.

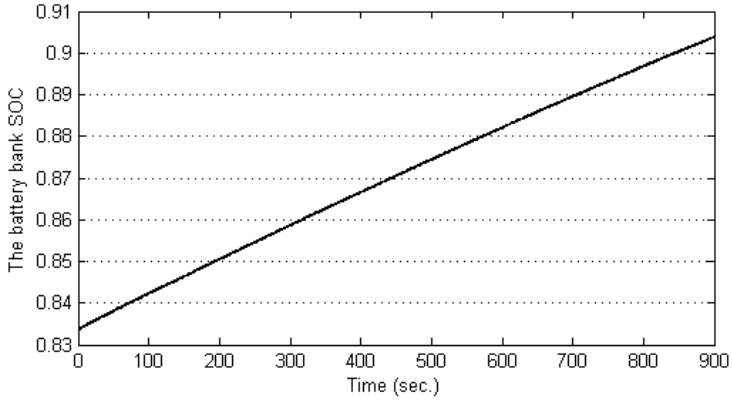
Figure 6.9 depicts the calculated optimal control variables for non-manipulated inputs in Figure 6.4. Moreover, Figures 6.10a-6.10d, respectively, illustrate dc bus voltage, C_p and angular velocity of wind turbine, as well as PV array current variations during the period of the operation time.



(a)

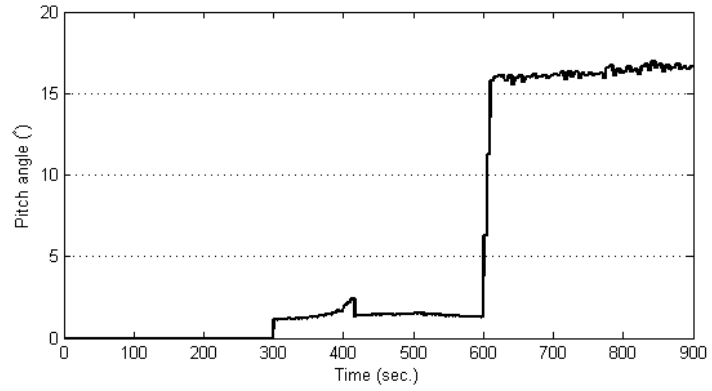


(b)

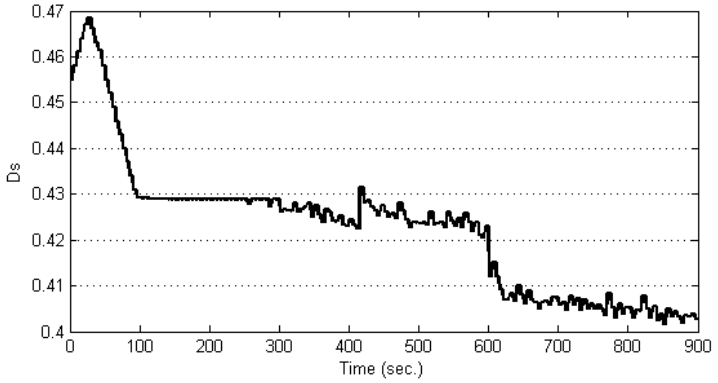


(c)

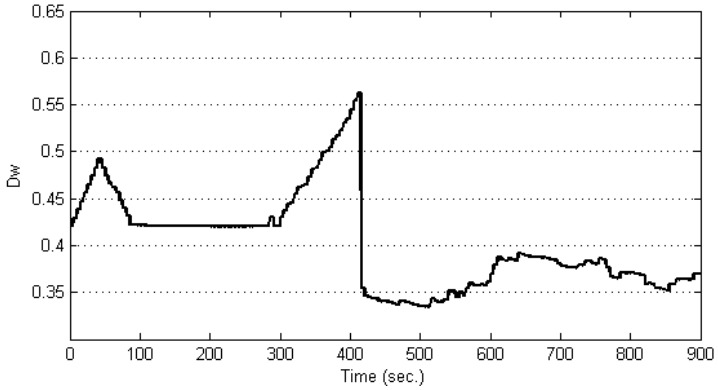
Figure 6.8: (a) Charging current, (c) terminal voltage, and (c) SOC of battery bank.



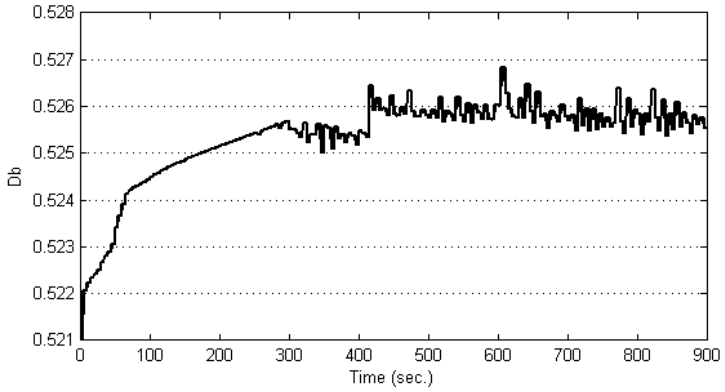
(a)



(b)

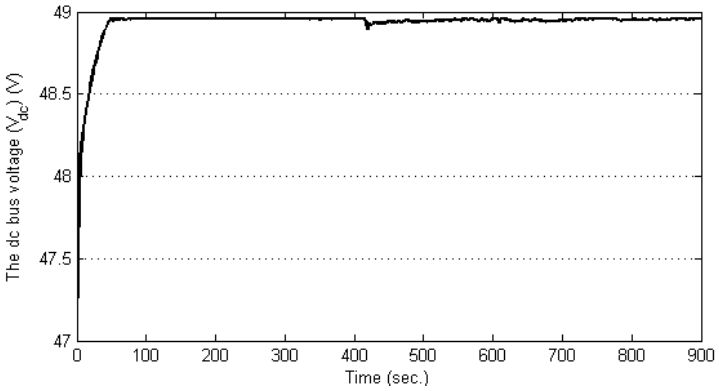


(c)

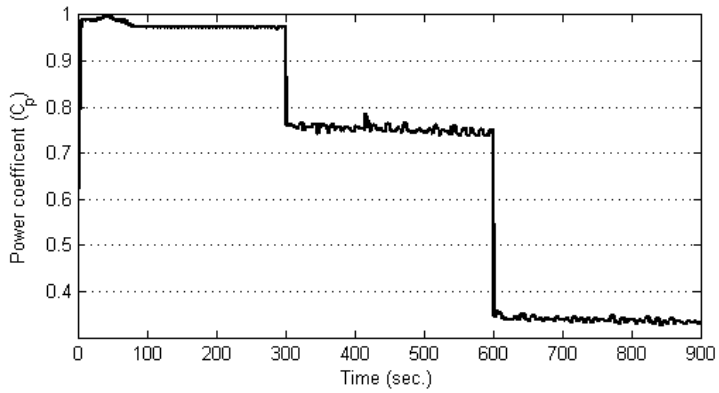


(d)

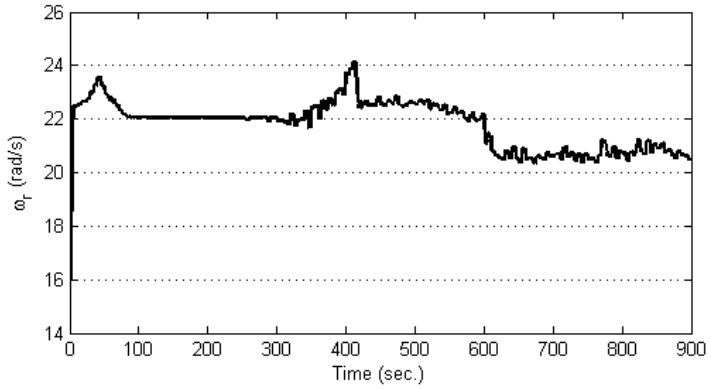
Figure 6.9: Optimal (a) pitch angle, switching duty cycles of (b) solar, (c) wind, and (d) battery branches.



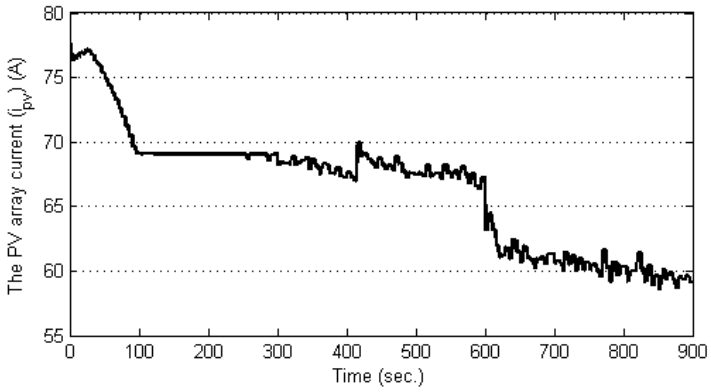
(a)



(b)



(c)



(d)

Figure 6.10: (a) Dc bus voltage level, (b) C_p, and (c) angular velocity of wind turbine, and (d) PV array current.

In spite of a significant variations of wind speed, Figure 6.10a shows that dc bus voltage of microgrid stays constant within the permissible range, which is $\overline{V}_{dc} \pm 0.02\overline{V}_{dc}$. From Figure 6.10a, it can be seen that dc bus voltage is maintained at the maximum possible level to store the maximum possible excess of generation.

Wind branch operates close to its maximum power point before $t = 300$ seconds to supply the sum of the load demand as well as the charging power. The resulting rotational speed of wind turbine is close to its nominal value as given by Figure 6.10c leading to produce close to unity power coefficient (Figure 6.10b). In order to maintain the battery bank voltage level, generated power is curtailed after $t = 300$ and $t = 600$ seconds by increasing pitch angle and then reducing rotational speed as given in Figure 6.10c. As a result, the power coefficient is declined to 0.75 and then 0.33 (Figure 6.10b).

Figure 6.10d, on the other hand, illustrates that during the first stage, solar branch also works close to its MPP, i.e. $I_{pv} = 76.1A$; however, its generated power is curtailed after $t = 300$ seconds when controller changes its operation point in order to curtail its generated power.

6.5 Summary

In this chapter, a novel optimal EMS was developed to manage energy flows across a sample of stand-alone green dc microgrids, consisting of wind, solar, and battery branches. This optimal EMS is a coordinated and multivariable NMPC strategy that addresses three main control objectives of stand-alone dc microgrids. These objectives are voltage level regulation, proportional power sharing, and battery management. In order to address these objectives, the developed EMS simultaneously controls pitch angle of wind turbine as well as switching duty cycles of three dc-dc converters. It has been shown that the developed controller tracks MPPs of wind and solar branches within normal conditions and curtails their generations during under-load conditions. The provided flexible generation curtailment strategy realizes a constant current-constant voltage charging regime for the connected battery bank that potentially increases its life span. It is important to note that the proposed strategy can be a centralized implementation of the primary and secondary levels of the hierarchical architecture. The obtained simulation results have shown the ability

of the developed controller to achieve all control objectives. The issue of considering discharging mode of battery operation, which shifts the problem to the class of hybrid dynamical systems, is currently being investigated.

Chapter 7

Conclusion

This dissertation work comprised a study on modelling, simulation and control of stand-alone dc microgrids for the purpose of developing dynamic energy management strategies. It was shown that such the energy management strategies are non-linear multivariable optimal control problems. This dissertation work involved formulating such the optimal control problems and developing non-linear model predictive control strategies to optimally manage energy flows across stand-alone dc microgrids.

This chapter summarizes main contributions of the work. It also addresses some open issues and point directions for further research.

7.1 Original contributions

This thesis contributes to the following domains:

- Mathematical modelling of stand-alone dc microgrids as hybrid systems;
- Model identification of PV module and energy resources;
- Acausal long-term simulation of stand-alone dc microgrids;
- Optimal energy management strategies for stand-alone dc microgrids;

The obtained results are discussed in more detail throughout the following subsections.

7.1.1 Mathematical modelling of stand-alone dc microgrids as hybrid systems

The available algebraic constraints and bimodal operation of battery banks cause the mathematical model of stand-alone dc microgrids to be a set of hybrid differential algebraic equations (hybrid DAEs). Moreover, the cycle life state of batteries, as a discontinuous differential state, make the system to be of non-Filippov type. In order to develop a mathematical model applicable to NMPC strategies, stand-alone dc microgrids were approximated as complementarity systems (CSs) of Filippov type, including mixed complementarity problems. The introduced mathematical model was used in Chapters 5 and 6, respectively, for long-term simulation and development of the optimal energy management strategies for stand-alone dc microgrids.

7.1.2 Model identification of PV module and energy resources

As part of the modelling efforts, stochastic models of hourly wind speed and solar irradiance were presented to simulate these renewable energy resources for locations across the UK. The presented stochastic models were assessed, using Kolmogrov-Smirnov (K-S) criteria, against available meteorological data.

Furthermore, a novel algorithm was proposed to identify an accurate equivalent electrical circuit for PV modules. In spite of the prior PV parameter identification methods, the presented algorithm uses both STC and NOCT information provided by manufacturers. The proposed algorithm constructs a non-linear optimization problem, based on the main algebraic equation of PV modules and available data in datasheets, to identify the electrical parameters of an equivalent circuit. It was shown that the overall equivalent circuit error, which is defined as the sum of the MPP errors of both STC and NOCT conditions, is improved by a factor between 5.1% to 31% depending on underlying PV technology.

7.1.3 Acausal long-term simulation of stand-alone dc microgrids

Long-term simulation of stand-alone dc microgrids is becoming essential to predict their behaviours under wind speed and solar irradiance uncertainties. A sample stand-alone dc microgrid, based on the proposed mathematical model of the system, is simulated for both its fast and slow dynamics. A Matlab/SimPowerSystems

acausal model was developed in order to simulate all fast and slow dynamics for a short period of simulation time. On the other hand, Modelica, as a declarative modelling language, was used to develop a model of the system for long-term simulation purpose. OpenModelica platform was also used to compile the developed Modelica model, as well as to simulate it for long period of simulation time.

7.1.4 Optimal energy management strategies for stand-alone dc microgrids

The classical energy management strategies employ MPPT algorithms and rely on batteries to absorb any possible excess energy. However, in order to realize constant current-constant voltage (IU) charging regime and increase the life span of batteries, energy management strategies require being more flexible, equipped with a power curtailment feature. In this thesis, a NMPC-based coordinated and multivariable EMS was developed to control the operation of a stand-alone DC microgrid. The proposed strategy employs wind and solar branches as controllable generators in order to achieve five control objectives: i) supplying variable dc loads; ii) regulating dc bus voltage level; iii) charging batteries as close to IU regime as possible; iv) proportional power sharing between generators; v) tracking MPPs of wind and PV branches during their normal operations. It was shown that applying the proposed controller to a sample stand-alone dc microgrid, battery bank is charged according to the IU charging regime. A variable load demand is also shared accurately between generators in proportion to their ratings. Moreover, dc bus voltage is maintained at its nominal level with the maximum error of 2.0%.

7.2 Critical appraisal and future works

The techniques and results, presented throughout this thesis, can be possibly further improved as the objectives of the following future research directions.

In order to practically deploy the developed NMPC strategies to stand-alone dc microgrids, stability of the closed-loop system must be analyzed. According to available stability theories for NMPC strategies, it needs to calculate appropriate stabilizing terminal penalty term and terminal region constraint to be added to the original NMPC. However, it needs to re-evaluate available theories for the class of MPCCs.

Although the presented NMPC strategy for the class of MPCCs was simplified to

be solved with NLP solvers, the original problem may be solvable by using adaptive penalization of the complementarity constraints or by employing mixed integer nonlinear programming solvers.

The proposed optimal energy management strategies were developed and simulated in Python. it is very interesting to validate the obtained results with the results of a Hardware-in-the-Loop and experimental test environments. In order to deploy the developed strategies for a real stand-alone dc microgrid, it is interesting to improve its performance including the processing latency.

Moreover, the developed PV identification algorithm can be extended to identify the electrical parameters of an equivalent circuit with two or more diodes. Two-diode equivalent circuit of PV modules provides better modeling of the loss in depletion region caused by recombination of carriers.

Finally, it is important to investigate any possible correlation between wind speed and solar irradiance, as two stochastic processes. The correlation between wind speed and insolation can be modeled as a joint probability distribution which is used for coordinated simulation of these uncertain sources of energy.

Appendix A

List of Publications

- A. M. Dizqah, A. Maheri, K. Busawon, P. Fritzson, "Standalone DC Microgrids as Complementarity Dynamical Systems: Modeling and Applications," Control Engineering and Practice (under review)
- A. M. Dizqah, A. Maheri, K. Busawon, and P. Fritzson, "Acausal Modelling and Dynamic Simulation of the Standalone WindSolar Plant Using Modelica," in IEEE 15th International Conference on Computer Modelling and Simulation (UKSim2013), (Cambridge, UK), pp. 580-585, April 2013
- A. M. Dizqah, K. Busawon, and P. Fritzson, "Acausal Modeling and Simulation of the Standalone Solar Power Systems as Hybrid DAEs," in The 53rd Int'l Conf. of the Scandinavian Simulation Sociey (SIMS2012), 2012
- A. Kamjoo, A. Maheri, G. A. Putrus, and A. M. Dizqah, "Optimal Sizing of Grid-Connected Hybrid Wind-PV Systems with Battery Bank Storage," in World Renewable Energy Forum (WREF), 2012
- A. Kamjoo, A. Maheri, A. M. Dizqah, and G. A. Putrus, "Multi-Objective Design Under Uncertainties of Standalone Hybrid Renewable Energy System Using NSGAII and Chance Constrained Programming," Energy (under review), 2013
- A. M. Dizqah, A. Maheri, and K. Busawon, "An Assessment of Wind Speed Stochastic Model for the UK," in 2nd International Symposium on Environment Friendly Energies and Applications (EFEA), Newcastle upon Tyne, 2012
- A. M. Dizqah, A. Maheri, and K. Busawon, "An Assessment of Solar Irradiance

Stochastic Model for the UK," in 2nd International Symposium on Environment Friendly Energies and Applications (EFEA), Newcastle upon Tyne, 2012

- A. M. Dizqah, A. Maheri, and K. Busawon, "An Accurate Method for the PV Model Identification Based on a Genetic Algorithm and the Interior-Point Method," Renewable Energy (accepted), 2014
- A. M. Dizqah, A. Maheri, K. Busawon, and A. Kamjoo, "Modelling and Simulation of Standalone Solar Power Systems," International Journal of Computational Methods and Experimental Measurements, (in press)
- A. M. Dizqah, A. Maheri, K. Busawon, and P. Fritzson, "Modeling and Simulation of a Combined Solar and Wind Energy System using OpenModelica," in 5th OpenModelica / 7th MODPROD Annual Workshops, (Linköping, Sweden), 2013
- A. M. Dizqah, A. Maheri, K. Busawon, A. Kamjoo, "A Multivariable Optimal Energy Management Strategy for the Standalone DC Microgrids," IEEE Transactions on Power Systems (under review)
- A. M. Dizqah, A. Maheri, K. Busawon, "Standalone Wind-Solar Battery Charger: A Nonlinear Predictive Approach," in European Control Conference (ECC2014) (under review), 2014

References

[1] J. M. Guerrero, M. Chandorkar, T. Lee, and P. C. Loh, “Advanced Control Architectures for Intelligent Microgrids–Part I: Decentralized and Hierarchical Control,” *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 4, pp. 1254–1262, 2013.

[2] P. Nema, R. K. Nema, and S. Rangnekar, “A current and future state of art development of hybrid energy system using wind and PV-solar: A review,” *Renewable and Sustainable Energy Reviews*, vol. 13, no. 8, pp. 2096 – 2103, 2009.

[3] R. S. Balog, W. W. Weaver, and P. T. Krein, “The Load as an Energy Asset in a Distributed DC SmartGrid Architecture,” *Smart Grid, IEEE Transactions on*, vol. 3, no. 1, pp. 253–260, 2012.

[4] D. Chen, and L. Xu, “Autonomous DC Voltage Control of a DC Microgrid With Multiple Slack Terminals,” *Power Systems, IEEE Transactions on*, vol. 27, no. 4, pp. 1897–1905, 2012.

[5] S. Anand, B. G. Fernandes, and M. Guerrero, “Distributed Control to Ensure Proportional Load Sharing and Improve Voltage Regulation in Low-Voltage DC Microgrids,” *Power Electronics, IEEE Transactions on*, vol. 28, no. 4, pp. 1900–1913, 2013.

[6] J. M. Guerrero, P. C. Loh, T. L. Lee, and M. Chandorkar, “Advanced Control Architectures for Intelligent Microgrids–Part II: Power Quality, Energy Storage, and AC/DC Microgrids,” *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 4, pp. 1263–1270, 2013.

[7] N. Eghtedarpour, and E. Farjah, "Control strategy for distributed integration of photovoltaic and energy storage systems in DC micro-grids," *Renewable Energy*, vol. 45, no. 0, pp. 96 – 110, 2012.

[8] X. Yu, and H. Qu, "Wind power in China–Opportunity goes with challenge," *Renewable and Sustainable Energy Reviews*, vol. 14, no. 8, pp. 2232 – 2237, 2010.

[9] T. Burton, N. Jenkins, D. Sharpe, and E. Bossanyi, *Wind Energy Handbook*. West Sussex, UK: John Wiley & Sons, 2 ed., 2011.

[10] A. Meharrar, M. Tioursi, M. Hatti, and A. B. Stambouli, "A variable speed wind generator maximum power tracking based on adaptative neuro-fuzzy inference system," *Expert Systems with Applications*, vol. 38, no. 6, pp. 7659 – 7664, 2011.

[11] C. T. Pan, and Y. L. Juan, "A Novel Sensorless MPPT Controller for a High-Efficiency Microscale Wind Power Generation System," *Energy Conversion, IEEE Transactions on*, vol. 25, pp. 207 –216, march 2010.

[12] H. Li, and Z. Chen, "Overview of different wind generator systems and their comparisons," *Renewable Power Generation*, vol. 2, no. 2, pp. 123–138, 2008.

[13] N. Mohan, T. M. Undeland, and W. P. Robbins, *Power electronics: converters, applications, and design*. New York: John Wiley & Sons, 2 ed., 1995.

[14] J. H. Su, J. J. Chen, and D. S. Wu, "Learning Feedback Controller Design of Switching Converters Via MATLABSIMULINK," *IEEE Transactions on Education*, vol. 45, pp. 307–315, 2002.

[15] R. D. Middledbrook,and S. Cuk, "A General Unified Approach to Modelling Switching-Converter Power Stages," in *Proc. of IEEE Power Electronics Specialist conference*, 1976.

[16] S. Mariethoz, S. Almer, M. Baja, A. G. Beccuti, D. Patino, A. Wernrud, J. Buisson, H. Cormerais, T. Geyer, H. Fujioka, U. T. Jonsson, C. Y. Kao, M. Morari, G. Papafotiou, A. Rantzer, P. Riedinger, "Comparison of Hybrid Control Techniques for Buck and Boost DC-DC converters," *IEEE Transactions on Control Systems Technology*, vol. 18, pp. 1126–1145, 2010.

[17] M. Beaudin, H. Zareipour, A. Schellenberglabe, and W. Rosehart, "Energy storage for mitigating the variability of renewable electricity sources: An updated review," *Energy for Sustainable Development*, vol. 14, pp. 302–314, 2010.

[18] K. C. Divya, and J. Ostergaard, "Battery energy storage technology for power systems - An overview," *Electric Power Systems Research*, vol. 79, pp. 511–520, 2009.

[19] D. Guasch, and S. Silvestre, "Dynamic Battery Model for Photovoltaic Applications," *Progress in Photovoltaics: Research and Applications*, vol. 11, pp. 193–206, 2003.

[20] O. Tremblay ,and L. Dessaint, "Experimental Validation of a Battery Dynamic Model for EV Applications," *World Electric Vehicle Journal*, vol. 3, pp. 10–15, 2009.

[21] H. Fakham, D. Lu, and B. Francois, "Power Control Design of a Battery Charger in a Hybrid Active PV Generator for Load-Following Applications," *Industrial Electronics, IEEE Transactions on*, vol. 58, pp. 85 –94, jan. 2011.

[22] P. Fritzson, *Introduction to modelling and Simulation of Technical and Physical Systems with Modelica*. New York: John Wiley & Sons, 2011.

[23] P. Fritzson, *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. Wiley, 2004.

[24] R. E. Cellier, and E. Kofman, *Continuous System Simulation*. Springer, 2006.

[25] L. T. Biegler, *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*. SIAM e-books, Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104), 2010.

[26] P. Fritzson, *A Quick Tour of Modelica*, pp. 19–69. John Wiley & Sons, Inc., 2004.

[27] J. Åkesson, K. E. Årzén, M. Gäfvert, T. Bergdahl, and H. Tummescheit, "Modeling and optimization with Optimica and JModelica.org–Languages and tools

for solving large-scale dynamic optimization problems," *Computers & Chemical Engineering*, vol. 34, no. 11, pp. 1737 – 1749, 2010.

[28] P. Fritzson, P. Aronsson, H. Lundvall, K. Nyström, A. Pop, L. Saldamli, and D. Broman, "The OpenModelica Modeling, Simulation, and Development Environment," 2005.

[29] M. Lazar, *Model predictive control of hybrid systems: Stability and robustness*. PhD thesis, Eindhoven University of Technology, Netherland, 2006.

[30] R. Goebel, R. G. Sanfelice, and A. R. Teel, "Hybrid dynamical systems," *Control Systems, IEEE*, vol. 29, no. 2, pp. 28–93, 2009.

[31] W. P. M. H. Heemels, and B. Brogliato, "The Complementarity Class of Hybrid Dynamical Systems," *European Journal of Control*, vol. 9, no. 2-3, pp. 322 – 360, 2003.

[32] A. F. Filippov, and F. M. Arscott, *Differential Equations with Discontinuous Righthand Sides: Control Systems*. Mathematics and its Applications, Kluwer Academic Publishers, 1988.

[33] W. S. Lin, and C. H. Zheng, "Energy management of a fuel cell/ultracapacitor hybrid power system using an adaptive optimal-control method," *Journal of Power Sources*, vol. 196, no. 6, pp. 3280 – 3289, 2011.

[34] Y. Haitao, Z. Yulan, S. Zechang, and W. Gang, "Model-based power control strategy development of a fuel cell hybrid vehicle," *Journal of Power Sources*, vol. 180, no. 2, pp. 821 – 829, 2008.

[35] S. J. Moura, D. S. Callaway, H. K. Fathy, and J. L. Stein, "Tradeoffs between battery energy capacity and stochastic optimal power management in plug-in hybrid electric vehicles," *Journal of Power Sources*, vol. 195, no. 9, pp. 2979 – 2988, 2010.

[36] D. Feroldi, M. Serra, and J. Riera, "Energy Management Strategies based on efficiency map for Fuel Cell Hybrid Vehicles," *Journal of Power Sources*, vol. 190, no. 2, pp. 387 – 401, 2009.

[37] A. Nosrat, and J. M. Pearce, "Dispatch strategy and model for hybrid photovoltaic and trigeneration power systems," *Applied Energy*, vol. 88, no. 9, pp. 3270 – 3276, 2011.

[38] H. Beltran, E. Bilbao, E. Belenguer, I. Etxeberria-Otadui, and P. Rodriguez, "Evaluation of Storage Energy Requirements for Constant Production in PV Power Plants," *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 3, pp. 1225–1234, 2013.

[39] D. Ipsakis, S. Voutetakis, P. Seferlis, F. Stergiopoulos, and C. Elmasides, "Power management strategies for a stand-alone power system using renewable energy sources and hydrogen storage," *International Journal of Hydrogen Energy*, vol. 34, no. 16, pp. 7081 – 7095, 2009.

[40] S. H. Karaki, R. B. Chedid, and R. Ramadan, "Probabilistic performance assessment of autonomous solar-wind energy conversion systems," *Energy Conversion, IEEE Transactions on*, vol. 14, pp. 766 –772, sep 1999.

[41] M.Y. Sulaiman, W.M. H. Oo, M. A. Wahab, and A. Zakaria, "Application of beta distribution model to Malaysian sunshine data," *Renewable Energy*, vol. 18, no. 4, pp. 573 – 579, 1999.

[42] A. Mellit, H. Mekki, A. Messai, and S.A. Kalogirou, "FPGA-based implementation of intelligent predictor for global solar irradiation, Part I: Theory and simulation," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2668 – 2685, 2011.

[43] A. Mefti, M. Y. Bouroubi, and A. Adane, "Generation of hourly solar radiation for inclined surfaces using monthly mean sunshine duration in Algeria," *Energy Conversion and Management*, vol. 44, no. 19, pp. 3125 – 3141, 2003.

[44] S. S. Sanz, A. M. P. Bellido, E. G. O. Garcia, A. P. Figueras, L. Prieto, and F. Correoso, "Accurate short-term wind speed prediction by exploiting diversity in input data using banks of artificial neural networks," *Neurocomputing*, vol. 72, no. 4-6, pp. 1336 – 1341, 2009.

[45] W. G. Fruh, "Long-term wind resource and uncertainty estimation using wind records from Scotland as example," *Renewable Energy*, vol. 50, no. 0, pp. 1014 – 1026, 2013.

[46] M. G. Villalva, J. R. Gazoli, and E. R. Filho, "Comprehensive Approach to Modeling and Simulation of Photovoltaic Arrays," *IEEE Transactions on Power Electronics*, vol. 24, pp. 1198–1208, 2009.

[47] M. K. Deshmukh, and S. S. Deshmukh, "Modelling of Hybrid Renewable Energy System," *Renewable And Sustainable Energy Reviews*, vol. 12, pp. 235–249, 2008.

[48] J. Martínez, and A. Medina, "A State Space Model for the Dynamic Operation Representation of Small-Scale Wind-Photovoltaic Hybrid Systems," *Journal of Renewable Energy*, vol. 35, pp. 1159–1168, 2010.

[49] M. Ceraolo, "New dynamical models of lead-acid batteries," *Power Systems, IEEE Transactions on*, vol. 15, pp. 1184 –1190, nov 2000.

[50] A. Gupta, R.P. Saini, and M.P. Sharma, "Modelling of hybrid energy system: Part II: Combined dispatch strategies and solution algorithm," *Renewable Energy*, vol. 36, no. 2, pp. 466 – 473, 2011.

[51] X. Hu, S. Li, and H. Peng, "A comparative study of equivalent circuit models for Li-ion batteries," *Journal of Power Sources*, vol. 198, no. 0, pp. 359 – 367, 2012.

[52] K. Mamadou, T. M. P. Nguyen, E. L. Potteau, C. Glaize, and J. Alzieu, "New accelerated charge methods using early destratification applied on flooded lead acid batteries," *Journal of Power Sources*, vol. 196, no. 8, pp. 3981 – 3987, 2011.

[53] F. J. Lin, M. S. Huang, P. Y. Yeh, H. C. Tsai, and C. H. Kuan, "DSP-Based Probabilistic Fuzzy Neural Network Control for LiIon Battery Charger," *Power Electronics, IEEE Transactions on*, vol. 27, pp. 3782 –3794, aug. 2012.

[54] R. Carter, A. Cruden, and P. J. Hall, "Optimizing for Efficiency or Battery Life in a Battery/Supercapacitor Electric Vehicle," *Vehicular Technology, IEEE Transactions on*, vol. 61, pp. 1526 –1533, may 2012.

[55] F. Locment, M. Sechilariu, and I. Houssamo, "DC Load and Batteries Control Limitations for Photovoltaic Systems Experimental Validation," *Power Electronics, IEEE Transactions on*, vol. 27, pp. 4030 –4038, sept. 2012.

[56] S. Grillo, M. Marinelli, S. Massucco, and F. Silvestro, "Optimal Management Strategy of a Battery-Based Storage System to Improve Renewable Energy Integration in Distribution Networks," *Smart Grid, IEEE Transactions on*, vol. 3, pp. 950 –958, june 2012.

[57] F. Valenciaga, and P. F. Puleston, “Supervisor control for a stand-alone hybrid generation system using wind and photovoltaic energy,” *Energy Conversion, IEEE Transactions on*, vol. 20, pp. 398 – 405, june 2005.

[58] H. Ghoddami, M. B. Delghavi, and A. Yazdani, “An integrated wind-photovoltaic-battery system with reduced power-electronic interface and fast control for grid-tied and off-grid applications,” *Renewable Energy*, vol. 45, no. 0, pp. 128 – 137, 2012.

[59] X. Liu, and W. Xu, “Economic Load Dispatch Constrained by Wind Power Availability: A Here-and-Now Approach,” *Sustainable Energy, IEEE Transactions on*, vol. 1, pp. 2 –9, april 2010.

[60] M. Khalid, and A.V. Savkin, “A model predictive control approach to the problem of wind power smoothingÂ with controlled battery storage,” *Renewable Energy*, vol. 35, no. 7, pp. 1520 – 1526, 2010.

[61] W. H. Kwon, and S. H. Han, “Optimal Controls on Finite and Infinite Horizons: A Review,” in *Receding Horizon Control*, Advanced Textbooks in Control and Signal Processing, pp. 17–82, Springer London, 2005.

[62] L. Grüne, and J. Pannek, *Nonlinear Model Predictive Control: Theory and Algorithms*. Communications and control engineering, Springer, 2011.

[63] R. Findeisen, and F. Allgöwer, “An Introduction to Nonlinear Model Predictive,” in *Control, 21st Benelux Meeting on Systems and Control, Veidhoven*, pp. 1–23, 2002.

[64] L. Grüne, “NMPC without terminal constraints,” in *4th IFAC Nonlinear Model Predictive Control Conference, Leeuwenhorst, Netherlands*, International Federation of Automatic Control, 2012.

[65] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789 – 814, 2000.

[66] V. Zavala, C. D. Laird, and L. T. Biegler, “Fast implementations and rigorous models: Can both be accommodated in NMPC?,” *International Journal of Robust and Nonlinear Control*, vol. 18, no. 8, pp. 800–815, 2008.

[67] E. F. Camacho, and C. Bordons, *Model Predictive Control*. Advanced Textbooks in Control and Signal Processing Series, SpringerVerlag GmbH, 2004.

[68] L.T. Biegler, and V.M. Zavala, "Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization," *Computers & Chemical Engineering*, vol. 33, no. 3, pp. 575 – 582, 2009.

[69] M. Mahmood, and P. Mhaskar, "Lyapunov-based model predictive control of stochastic nonlinear systems," *Automatica*, vol. 48, no. 9, pp. 2271 – 2276, 2012.

[70] M. Diehl, R. Amrit, and J. B. Rawlings, "A Lyapunov Function for Economic Optimizing Model Predictive Control," *Automatic Control, IEEE Transactions on*, vol. 56, no. 3, pp. 703–707, 2011.

[71] M. Heidarinejad, J. Liu, and P. D. Christofides, "Economic model predictive control of switched nonlinear systems," *Systems & Control Letters*, vol. 62, no. 1, pp. 77 – 84, 2013.

[72] S. Boyd,L. Vandenberghe, *Convex Optimization*. New York: Cambridge University Press, 2004.

[73] B. Lincoln, and A. Rantzer, "Relaxing dynamic programming," *Automatic Control, IEEE Transactions on*, vol. 51, no. 8, pp. 1249–1260, 2006.

[74] N. Kim, S. Cha, and H. Peng, "Optimal Control of Hybrid Electric Vehicles Based on Pontryagin's Minimum Principle," *Control Systems Technology, IEEE Transactions on*, vol. 19, no. 5, pp. 1279–1287, 2011.

[75] C. Kirches, L. Wirsching, H.G. Bock, and J.P. Schlöder, "Efficient direct multiple shooting for nonlinear model predictive control on long horizons," *Journal of Process Control*, vol. 22, no. 3, pp. 540 – 550, 2012.

[76] R. A. Bartlett, A. Wachter, and L. T. Biegler, "Active set vs. interior point strategies for model predictive control," in *American Control Conference, 2000. Proceedings of the 2000*, vol. 6, pp. 4229–4233 vol.6, 2000.

[77] J. Andersson, J. Åkesson, and M. Diehl, "Dynamic optimization with CasADi," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pp. 681–686, 2012.

[78] R. Neidinger, “Introduction to Automatic Differentiation and MATLAB Object-Oriented Programming,” *SIAM Review*, vol. 52, no. 3, pp. 545–563, 2010.

[79] A. Verma, “An introduction to automatic differentiation,” *Current Science*, vol. 78, pp. 804–807, Apr. 2000.

[80] A. Griewank, and A. Walther, “Introduction to Automatic Differentiation,” *PAMM*, vol. 2, no. 1, pp. 45–49, 2003.

[81] J. Andersson, J. Åkesson, and Moritz Diehl, “CasADi – A Symbolic Package for Automatic Differentiation and Optimal Control,” in *Recent Advances in Algorithmic Differentiation* (S. Forth, P. Hovland, E. Phipps, J. Utke, and A. Walther, eds.), vol. 87 of *Lecture Notes in Computational Science and Engineering*, pp. 297–307, Springer Berlin Heidelberg, 2012.

[82] B. Houska, H. J. Ferreau, and M. Diehl, “An auto-generated real-time iteration algorithm for nonlinear $\{$MPC$\}$ in the microsecond range,” *Automatica*, vol. 47, no. 10, pp. 2279 – 2285, 2011.

[83] H. Bock, and K. Plitt, “A multiple shooting algorithm for direct solution of optimal control problems,” in *9th IFAC World Congress, Budapest*, pp. 243–247, Pergamon Press, 1984.

[84] H. Scheel, and S. Scholtes, “Mathematical Programs with Complementarity Constraints: Stationarity, Optimality, and Sensitivity,” *Mathematics of Operations Research*, vol. 25, no. 1, pp. 1–22, 2000.

[85] M. Anitescu, “On Using the Elastic Mode in Nonlinear Programming Approaches to Mathematical Programs with Complementarity Constraints,” *SIAM J. on Optimization*, vol. 15, pp. 1203–1236, Apr. 2005.

[86] A. J. van der Schaft, and J. M. Schumacher, *An Introduction to Hybrid Dynamical Systems*. Lecture Notes in Control and Information Sciences, Springer, 2000.

[87] D. E. Stewart, and M. Anitescu, “Optimal control of systems with discontinuous differential equations,” *Numerische Mathematik.*, vol. 114, pp. 653–695, Jan. 2010.

[88] J. J. Justo, F. Mwasilu, J. Lee, and J. W. Jung, "AC-microgrids versus DC-microgrids with distributed energy resources: A review," *Renewable and Sustainable Energy Reviews*, vol. 24, no. 0, pp. 387 – 405, 2013.

[89] S. Schuler, D. Schlipf, P. W. Cheng, and F. Allgöwer, "ℓ_1-Optimal Control of Large Wind Turbines," *Control Systems Technology, IEEE Transactions on*, vol. 21, no. 4, pp. 1079–1089, 2013.

[90] W. Qi, J. Liu, and P. D. Christofides, "Distributed Supervisory Predictive Control of Distributed Wind and Solar Energy Systems," *Control Systems Technology, IEEE Transactions on*, vol. 21, no. 2, pp. 504–512, 2013.

[91] L. Gkatzikis, I. Koutsopoulos, T. Salonidis, "The Role of Aggregators in Smart Grid Demand Response Markets," *IEEE Journal on Selected Areas in Communications*, vol. to apper, 2013.

[92] J. J. Soon, and K. S. Low, "Photovoltaic Model Identification Using Particle Swarm Optimization With Inverse Barrier Constraints," *IEEE Transactions on Power Electronics*, vol. 27, pp. 3975–3983, 2012.

[93] M. Ashari and C. Nayar, "An optimum dispatch strategy using set points for a photovoltaic (PV)-diesel-battery hybrid power system," *Solar Energy*, vol. 66, no. 1, pp. 1 – 9, 1999.

[94] S. Heier, *Grid Integration of Wind Energy Conversion Systems*. John Wiley & Sons, 1998.

[95] F. Y. Ettoumi, A. Mefti, A. Adane, and M. Y. Bouroubi, "Statistical analysis of solar measurements in Algeria using beta distributions," *Renewable Energy*, vol. 26, no. 1, pp. 47 – 67, 2002.

[96] H. Aksoy, Z. F. Toprak, A. Aytek, and N E. Unal, "Stochastic generation of hourly mean wind speed data," *Renewable Energy*, vol. 29, no. 14, pp. 2111 – 2131, 2004.

[97] G. Tina, and S. Gagliano, "Probabilistic analysis of weather data for a hybrid solar/wind energy system," *International Journal of Energy Research*, vol. 35, no. 3, pp. 221–232, 2011.

[98] G. W. Paltridge, and C. M. R. Platt, "Radiative Processes in Meteorology and Climatology," *Developments in Atmospheric Science*, vol. 5, 1976.

[99] U. M. Office, "MIDAS Land Surface Stations data (1853-current)." NCAS British Atmospheric Data Centre, 2006-2013.

[100] J. L. Agorreta, M. Borrega, J. Lopez, and L. Marroyo, "Modeling and Control of N-Paralleled Grid-Connected Inverters With LCL Fitler Coupled Due to Grid Impedance in PV Plants," *IEEE Transactions on Power Electronics*, vol. 26, no. 3, pp. 770–785, 2011.

[101] P. E. Kakosimos, and A. G. Kladas, "Implementation of Photovoltaic Array MPPT Through Fixed Step Predictive Control Technique," *Renewable Energy*, vol. 36, pp. 2508–2514, 2011.

[102] A. Kouchaki, H. Iman-Eini, B. Asaei, "A new maximum power point tracking strategy for {PV} arrays under uniform and nonuniform insolation conditions," *Solar Energy*, vol. 91, no. 0, pp. 221 – 232, 2013.

[103] K. Ishaque, Z. Salam, M. Amjad, and S. Mekhilef, "An Improved Particle Swarm Optimization (PSO)-Based MPPT for PV With Reduced Steady-State Oscillation," *IEEE Transactions on Power Electronics*, vol. 27, no. 8, pp. 3627–3638, 2012.

[104] T. Bennett, A. Zilouchian, R. Messenger, "A proposed maximum power point tracking algorithm based on a new testing standard," *Solar Energy*, vol. 89, no. 0, pp. 23 – 41, 2013.

[105] E. Karatepe, M. Boztepe, and M. Colak, "Development of a Suitable Model for Characterizing Photovoltaic Arrays with Shaded Solar Cells," *Journal of Solar Energy*, vol. 81, pp. 977–992, 2007.

[106] A. Mellit, and S. A. Kalogirou, "ANFIS-Based Modelling for Photovoltaic Power Supply System: A Case Study," *Journal of Renewable Energy*, vol. 36, pp. 250–258, 2011.

[107] J. A. Gow, and C. D. Manning, "Development of a Photovoltaic Array Model for Use in Power-Electronics Simulation Studies," in *Proc. IEE Electric Power Applications*, vol. 146, pp. 193–200, 1999.

[108] K. Ishaque, Z. Salam, and H. Taheri, “Simple, Fast and Accurate Two-Diode Model for Photovoltaic Modules,” *Journal of Solar Energy Materials and Solar Cells*, vol. 95, pp. 586–594, 2011.

[109] G. Walker, “Evaluating MPPT Converter Topologies Using a Matlab PV Model,” *Journal of Electrical and Electronics Engineering*, vol. 21, pp. 45–55, 2001.

[110] A. N. Celik, and N. Acikgoz, “Modelling and Experimental Verification of the Operating Current of Mono-Crystalline Photovoltaic Modules Using Four- and Five-Parameter Models,” *Journal of Applied Energy*, vol. 84, pp. 1–15, 2007.

[111] D. S. H. Chan, and J. C. H. Phang, “Analytical Methods for the Extraction of Solar-Cell Single- and Double-Diode Model Parameters from I-V Charactristics,” *IEEE Transactions on Electron Devices*, vol. ED-34, pp. 286–293, 1987.

[112] W. De Soto, S. A. Klein, and W. A. Beckman, “Improvement and Validation of a Model for Photovoltaic Array Performance,” *Journal of Solar Energy*, vol. 80, pp. 78–88, 2006.

[113] D. Sera, R. Teodorecu, and P. Rodriguez, “PV Panel Model Based on Datasheet Values,” in *Proc. IEEE Int. Symp. Electronic*, pp. 2392–2396, 2007.

[114] A. Chaterjee, A. Keyhani, and D. Kapoor, “Identification of Photovoltaic Source Models,” *IEEE Transactions on Energy Conversion*, vol. 80, pp. 78–88, 2011.

[115] R. Kardi, H. Andrei, J. P. Gaubert, T. Ivanovici, G. Champenois, and P. Andrei, “Modeling of the Photovoltaic Cell Circuit Parameters for Optimum Connection Model and Real-Time Enulator with Partial Shadow Conditions,” *Journal of Energy*, vol. 42, pp. 57–67, 2012.

[116] M. Zagrouba, A. Sellami, M. Bouaicha, and M. Ksouri, “Identification of PV Solar Cells and Modules Parameters Using the Genetic Algorithms: Application to Maximum Power Extraction,” *Journal of Solar Energy*, vol. 84, pp. 860–866, 2010.

[117] F. M. Petcut, and T. L. Dragomir, “Solar Cell Parameter Identification Using Genetic Algorithms,” *Journal of Control Engineering and Applied Informatics*, vol. 12, pp. 30–37, 2010.

[118] D. L. King, W. E. Boyson, and J. A. Kratochvil, "Photovoltaic Array Performance Model," Tech. Rep. SAND2004-3535, Sandia National Laboratories, 2004.

[119] U. Stutenbaeumer, and B. MEsfin, "Equivalent Model of Monocrystalline, Polycrystalline and Amorphous Silicon Solar Cells," *Journal of Renewable Energy*, vol. 18, pp. 501–512, 1999.

[120] SNL, "Sandia National Laboratories." www.sandia.gov.

[121] NIST, "National Institute of Standards and Technology." www.nist.gov.

[122] Q. Kou, S. A. Klein, and W. A. Beckman, "A Method for Estimating the Long-Term Performance of Direct-Coupled PV Pumping Systems," *Journal of Solar Energy*, vol. 64, pp. 33–40, 1998.

[123] R. T. Marler, and J. S. Arora, "Survey of Multi-Objective Optimization Methods for Engineering," *Journal of Structural Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, 2004.

[124] A. Konak, D. W. Coit, A. E. Smith, "Multi-Objective Optimization Using Genetic Algorithms: A Tutorial," *Journal of Reliability Engineering And System Safety*, vol. 91, pp. 992–1007, 2006.

[125] A. Wächter, and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[126] M. Gen, and R. Cheng, *Genetic Algorithms and Engineering Design*. New York: J. Wiley And Sons, 2000.

[127] O. Erdinc, and M. Uzunoglu, "Optimum design of hybrid renewable energy systems: Overview of different approaches," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 3, pp. 1412 – 1425, 2012.

[128] T. Kuo, and S. Y. Hwang, "A Genetic Algorithm with Disruptive Selection," *IEEE Transactions on Systems, Man, And Cybernetics*, vol. 26, no. 2, pp. 299–307, 1996.

[129] M. Rocha, and J. Neves, "Preventing Premature Convergence to Local Optima in Genetic Algorithms Via Random Offspring Generation," in *IEA/AIE '99*

Proceedings of the 12th Int'l. Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems: Multiple Approaches to Intelligent Systems, pp. 127–136, Springer-Verlag, 1999.

[130] Kyocera, "KC200GT, High Efficiency Multicrystal Photovoltaic Module." www.kyocerasolar.com/assets/001/5195.pdf, 2012.

[131] SHELL, "Shell ST40 Photovoltaic Solar Module." www.oeko-energie.de/prospekte/ShellST40_D.pdf, 2012.

[132] Sunpower, "High Efficiency Home Solar Panels - E20 Series." www.sunpowercorp.co.uk/homes/products-services/solar-panels/e20, 2012.

[133] I. H. Altas, and A. M. Sharaf, "A Photovoltaic Array Simulation Model for Matlab-Simulink GUI Environment," in *Proc. Intl. Conf. Clean Elect. Power (ICCEP)*, pp. 341–345, 2007.

[134] M. Buresch, *Photovoltaic Energy Systems Design and Installation*. New York: McGraw-Hill, 1983.

[135] K. Ishaque, Z. Salam, and H. Taheri, "Accurate MATLAB Simulink PV System Simulator Based on a Two-Diode Model," *Journal of Power Electronics*, vol. 11, pp. 179–187, 2011.

[136] D. Grenier, L. A. Dessaint, O. Akhrif, Y. Bonnassieux, and B. Le Pioufle, "Experimental nonlinear torque control of a permanent-magnet synchronous motor using saliency," *Industrial Electronics, IEEE Transactions on*, vol. 44, no. 5, pp. 680–687, 1997.

[137] L. R. Petzold, "A description of DASSL: a differential/algebraic system solver," in *Scientific computing (Montreal, Quebec, 1982)*, pp. 65–68, New Brunswick, NJ: IMACS, 1983.

[138] L. Xu, and D. Chen, "Control and Operation of a DC Microgrid With Variable Generation and Energy Storage," *Power Delivery, IEEE Transactions on*, vol. 26, no. 4, pp. 2513–2522, 2011.

[139] B. Zhao, X. Zhang, J. Chen, C. Wang, and L. Guo, "Operation Optimization of Standalone Microgrids Considering Lifetime Characteristics of Battery Energy

Storage System," *Sustainable Energy, IEEE Transactions on*, vol. PP, no. 99, pp. 1–10, 2013.

[140] T. Zhou, and B. Francois, "Energy Management and Power Control of a Hybrid Active Wind Generator for Distributed Power Generation and Grid Integration," *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 1, pp. 95–104, 2011.

[141] X. Liu, P. Wang, and P. C. Loh, "A Hybrid AC/DC Microgrid and Its Coordination Control," *Smart Grid, IEEE Transactions on*, vol. 2, no. 2, pp. 278–286, 2011.

[142] H. Kanchev, D. Lu, F. Colas, V. Lazarov, and B. Francois, "Energy Management and Operational Planning of a Microgrid With a PV-Based Active Generator for Smart Grid Applications," *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 10, pp. 4583–4592, 2011.

[143] J. M.Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuña, and M. Castilla, "Hierarchical Control of Droop-Controlled AC and DC Microgrids–A General Approach Toward Standardization," *Industrial Electronics, IEEE Transactions on*, vol. 58, no. 1, pp. 158–172, 2011.

[144] P. H. Divshali, A. Alimardani, S. H. Hosseinian, and M. Abedi, "Decentralized Cooperative Control Strategy of Microsources for Stabilizing Autonomous VSC-Based Microgrids," *Power Systems, IEEE Transactions on*, vol. 27, no. 4, pp. 1949–1959, 2012.

[145] P. C. Loh, D. Li, Y. K. Chai, and F. Blaabjerg, "Autonomous Operation of Hybrid Microgrid With AC and DC Subgrids," *Power Electronics, IEEE Transactions on*, vol. 28, no. 5, pp. 2214–2223, 2013.

[146] T. L. Vandoorn, B. Meersman, L. Degroote, B. Renders, and L. Vandevelde, "A Control Strategy for Islanded Microgrids With DC-Link Voltage Control," *Power Delivery, IEEE Transactions on*, vol. 26, no. 2, pp. 703–713, 2011.

[147] T. L. Vandoorn, B. Meersman, J. D. M. De Kooning, and L. Vandevelde, "Analogy Between Conventional Grid Control and Islanded Microgrid Control Based on a Global DC-Link Voltage Droop," *Power Delivery, IEEE Transactions on*, vol. 27, no. 3, pp. 1405–1414, 2012.

[148] H. Kakigano, Y. Miura, and T. Ise, "Distribution Voltage Control for DC Microgrids Using Fuzzy Control and Gain-Scheduling Technique," *Power Electronics, IEEE Transactions on*, vol. 28, no. 5, pp. 2246–2258, 2013.

[149] P. C. Loh, and F. Blaabjerg, "Autonomous control of distributed storages in microgrids," in *Power Electronics and ECCE Asia (ICPE ECCE), 2011 IEEE 8th International Conference on*, pp. 536–542, 2011.

[150] A. Kamjoo, A. Maheri, and G. Putrus, "Wind Speed and Solar Irradiance Variation Simulation Using ARMA Models in Design of HybridWind-PVBattery System," *Journal of Chean Energy Technologies*, vol. 1, pp. 14–17, 2013.